

LA PRIMA RIVISTA EUROPEA DI SOFTWARE PER PERSONAL COMPUTER

PERSONAL SOFTWARE

SPECIALE
POCKET

ANNO 2 N. 6
MAGGIO 1983 L. 3.500

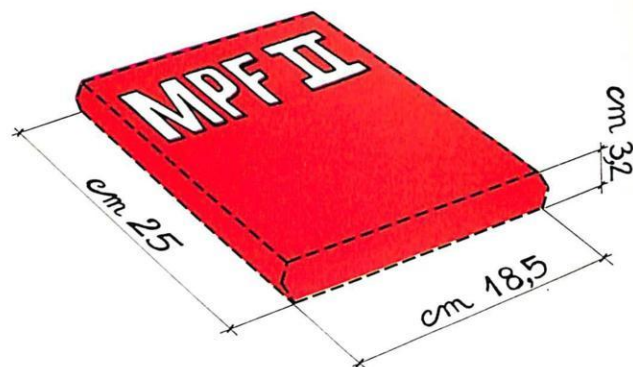


UNA PUBBLICAZIONE DEL GRUPPO EDITORIALE JACKSON



- **LOGICA ARISTOTELICA
CON FX-702P**
- **LA COMPRESSIONE
DEI DATI SU TI-59**
- **INTERFACCIAMENTO
TRA TRS-80 POCKET
E TRS-80 MOD. I**
- **MASTER MIND SU HP-29C**
- **QUADRATINI E LACUNE
PER TI-58 E TI-59**

1480 cm³



di **MICRO-PROFESSOR** **MPF II** contengono CPU R6502 - 64 K Bytes di RAM 16 K Bytes di ROM con Interprete Basic Apple Soft

Il MICROPROFESSOR II (MPFII) è un computer unico nel suo genere perché unisce a grandi capacità di memorie residenti (64 K Bytes di RAM e 16 K Bytes di ROM) una configurazione di sistema ridottissima. È veramente portatile.

Le sue minime dimensioni (cm 25 x 18,5 x 3,2) non gli impediscono però di essere un "personal computer" perché oltre ad essere dotato di eccezionali capacità di memoria residenti può essere completato ed allacciato con diverse periferiche.

MPFII diventa così un computer gestionale come altri computer più famosi ed "ingombranti" di lui.

Il modulatore RF e la scheda PALCOLOR residenti vi permetteranno di collegarlo al vostro televisore.

Ecco perché MPFII non è solo "lavoro", ma anche relax.

Insomma un computer idoneo per tutti, dai 7 ai 70 anni di età.

L'ampia disponibilità di software in cassetta, dischi e cartuccia (cartridge) costituisce l'elemento preponderante che lo rende indispensabile come: **SUPPORTO GESTIONALE** (amministrazione, magazzino, acquisti, commerciale, ecc.) per negozi, uffici, aziende. **SUPPORTO SCIENTIFICO PRATICO** per tecnici, professionisti, ricercatori, hobbysti. **SUPPORTO DIDATTICO** per studenti. **SUP-PORTO RICREATIVO** (giochi, quiz, ecc.) per tutti.



- 1) Computer
- 2) Interfaccia per disk drive
- 3) Disk drive (slim line)
- 4) Tastiera esterna

DIGITEK COMPUTER

Ufficio Vendite
Via Marmolada, 9/11 43058 SORBOLO (Parma)
Tel. 0521/69635 Telex 531083

non perdetevi la testa...



Rispondere alle esigenze sempre maggiori degli utilizzatori di mini e micro computers è la missione che si è fissata RHONE-POULENC SYSTEMES fabbricando FLEXETTE, i quali sono oggetto di controlli permanenti e in particolare d'un trattamento originale ed esclusivo di tutta la superficie del disco. Questa lavorazione consente di preservare le teste di registrazione, d'assicurare delle condizioni di lettura eccezionali ed aumentare la durata d'utilizzazione dei dischi. Non perdetevi più le Vostre teste di lettura, FLEXETTE le preserva ed assicura all'utilizzatore una manutenzione minima delle proprie attrezzature. Finalmente gli sforzi dei costruttori non sono più vani. FLEXETTE è riservato agli utilizzatori che ricercano la garanzia di un'alta tecnologia.

Per provare FLEXETTE nella Vostra regione:

 **RHÔNE
POULENC
SYSTEMES**
settore informatica
concessionari autorizzati

- | | |
|-----------|---|
| ♦ MILANO | - S.D.C. S.a.S. / Tel. 84.35.593 |
| ♦ TORINO | - PROGRAMMA UFFICIO S.a.S. / Tel. (011) 41.13.565 |
| ♦ VERONA | - MIDA S.r.l. / Tel. (045) 59.05.05 |
| ♦ FIRENZE | - C.S.S. S.n.c. / Tel. (055) 67.96.30 |
| ♦ PARMA | - TECNODATA S.a.S. / Tel. (0521) 25.079 |
| ♦ ROMA | - MASSIMO BRENUANI / Tel. (06) 81.27.665 |
| ♦ NAPOLI | - TES. IN / Tel. (081) 64.31.22 |
| ♦ BOLZANO | - DATAPLAN S.a.S. / Tel. (0471) 47.721-47.056 |

RHÔNE POULENC ITALIA S.p.A.
Divisione Rhône Poulenc Systemes
Via Romagnoli, 6 - 20146 MILANO tel. 42461
telex ITARPC 332330

Usare il sistema operativo CP/M

IL LIBRO

Il sistema operativo CP/M è stato progettato per rendere semplice l'uso di un microcomputer. Questo libro vi renderà semplice l'uso del CP/M. (Le versioni esaminate del CP/M sono il CP/M 1.4-il CP/M 2.2. e il nuovo sistema operativo multiutente MP/M) La maggior parte di utenti di microcomputer dovrà, infatti, un giorno o l'altro, fare ricorso al CP/M, disponibile su quasi tutti i computer basati sui microprocessori 8080 e Z80, come pure su certi sistemi utilizzando il 6502. Il libro, senza presupporre alcuna conoscenza di un calcolatore, inizia con la descrizione, passo-passo delle procedure di inizializzazione del sistema: accensione, inserimento dei dischetti, esecuzione delle più comuni operazioni su file, compresa la duplicazione dei dischetti. Prosegue con il PIP (programma di trasferimento dei file), il DDT (programma di messa a punto) e ED (programma editor). Per entrare sempre più, fornendo numerosi consigli pratici, all'interno del CP/M e delle sue operazioni, al fine di comprenderne appieno le risorse ed eventualmente dare gli strumenti per successive modifiche.

Pagg. 320 Cod. 510P
L. 22.000
(Abb. L. 19.800)



**GRUPPO EDITORIALE
JACKSON**
Divisione Libri

Tagliando d'ordine da inviare a:
Gruppo Editoriale Jackson - Via Rosellini, 12 - 20124 Milano

NOME _____

COGNOME _____

INDIRIZZO _____

CAP _____ CITTA' _____

Partita IVA (indispensabile per le aziende) _____

☐ Inviatemi il **Libro CP/M**

☐ Pagherò al postino l'importo indicato più L. 2.000
per le spese di spedizione.

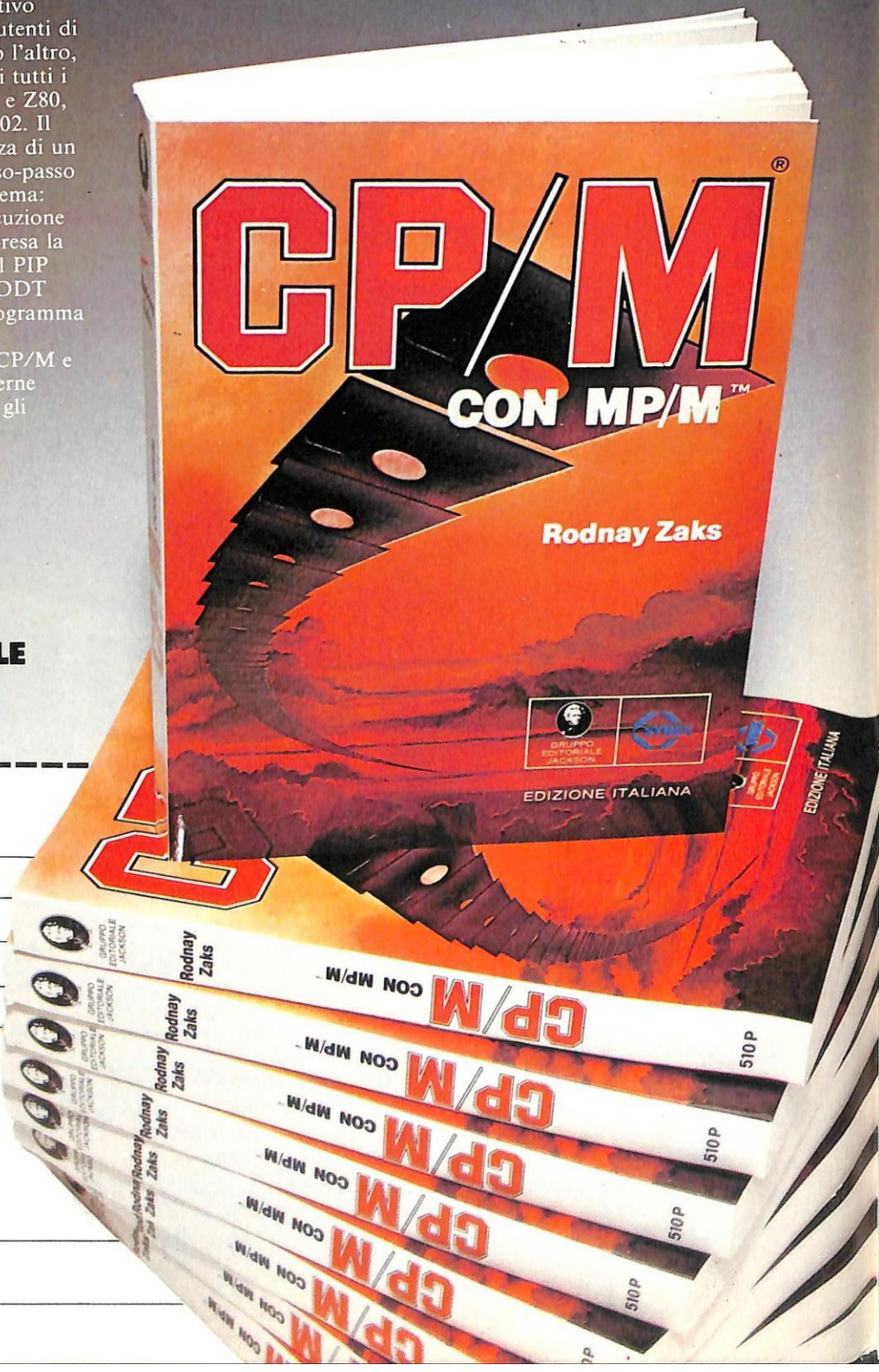
☐ Allego assegno n° _____

di L. _____

☐ Non Abbonato

☐ Abbonato (sconto 10%):
(indicare a quali riviste)

DATA _____ FIRMA _____



PERSONAL SOFTWARE

ARTICOLI

8	Un piccolo parallelo tra Pocket Basic	P. Capobussi....
16	Assolutizzare le etichette	P. Capobussi....
19	Operazioni, precisioni, approssimazioni	P. Capobussi....
21	Istogramma universale per PC 1211/TRS80 Pocket	Redazione....
22	Calcoli ingegneristici con TI 58/59	A. Scifoni....
28	Brevi ricette per ottimizzare	P. Capobussi....
31	Quadratini e lacune per TI 58 - 59	F. Carbone....
33	Tre programmi di fisica e chimica con l'HP67 - HP97	F. Sismondo....
39	La compressione dei dati	P. Capobussi....
42	Master Mind su HP - 29C	E. Bossaglia....
47	Interfacciamento: TRS 80 Pocket contro TRS 80 Mod 1	Redazione....
51	Moduli e fasi di funzioni di trasferimento con TI 59	G. Fedecostante....
55	Logica aristotelica con la calcolatrice tascabile	G. De Nicolao....

GUIDA

... tutti
... tutti
... tutti
... PC 1211/TRS Pocket
... TI 58/59
... tutti
... TI 58/59
... HP67/HP97
... tutti
... HP-29C
... TRS 80 Pocket
... TI 59
... FX 507

RUBRICHE

7	Editoriale	
	Il fascino dei computer	M. Boscarol....
62	Conversioni	
	Giochiamo d'azzardo sulle PC - 1500	E. Cima....
16	Piccoli Annunci	...PC 1500

Indirizzate tutta la corrispondenza editoriale a

Personal Software, Via Rosellini 12, 20124 Milano

I manoscritti non richiesti non saranno restituiti. Le opinioni espresse degli autori non sono necessariamente quelle di *Personal Software*.

È disponibile a richiesta una "Guida per gli autori" contenente le informazioni necessarie alla stesura di un articolo o di un programma, oltre a tutte le indicazioni di carattere amministrativo. Richiedetela all'indirizzo indicato.

In questa guida sono riportati i personal computer e i microprocessori di cui si parla negli articoli e nelle rubriche.

SAVING COMPUTER '83



La sorgente per le necessità del tuo computer

*Nella nostra sala mostra
potrai ammirare e provare prodotti come:*

- ☐ stampanti
- ☐ floppy disk
- ☐ programmi
- ☐ biblioteca specializzata

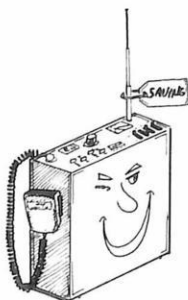
Le migliori marche di Personal Computer

*Disponiamo infatti pronta consegna
di APPLE II, APPLE II E, SIRIUS,
SORD M 23, AVT COMP 2, VIC 20,
VIC 64, ZX81, SPECTRUM, MPF II*

**Non perdere
questa occasione!!!**

**DISTRIBUTORI ESCLUSIVI DEL FAVOLOSO
"THE LAST ONE" PER IL VENETO**

**Vendita anche per corrispondenza,
telefona per le quotazioni, saremo lieti di accontentarti.**



SAVING

ELETTRONICA

VIA GRAMSCI 40 - MIRANO (VE) - TEL. (041) 432876

Il fascino dei computer

Mauro Boscarol

Ogni settimana ricevo decine di lettere dai lettori, appassionati ed hobbysti dell'informatica. Ed ogni volta, confesso, resto stupito. Mi scrivono ragazzi di dodici anni, professionisti, medici, architetti, studenti di tutti i livelli. Sono persone di diversa collocazione sociale e culturale; c'è chi sa tutto sull'informatica e chi è ai primi passi. Ma tutti scrivono ad una rivista specializzata in personal computer. Perché?

Per cercare di spiegarlo a me stesso, sono andato col pensiero ai miei primi contatti con i computer. Ripensando a quei tempi, mi sono chiesto cos'era che accendeva il mio entusiasmo. Ho ricordato il mio primo programma: la somma di due numeri in virgola mobile; mi è venuto da sorridere al pensiero di quel pacchettino di schede perforate, la maggior parte delle quali erano "schede controllo" e solo sei o sette erano schede di istruzioni assembler.

Quando mi consegnarono il risultato, mi persi in mezzo a quei tre fogli di tabulato pieni di numeri e sigle incomprensibili; finalmente, con l'aiuto di qualcuno più esperto di me, riuscii finalmente a trovare in un angolino il risultato.

Fu in quel momento che l'informatica mi conquistò. Essere riuscito a far calcolare ad un "cervello elettronico" che riempiva quattro stanze una cosa così piccola, ma tutta per me, e su miei comandi, mi entusiasmo come se mi avessero fatto pilotare per un attimo un Jumbo-Jet.

E d'improvviso mi apparvero tutte le possibili applicazioni, che allora non conoscevo, ma immaginavo esistessero, dei computer. Mi affascinava l'idea di riuscire ad insegnare qualcosa ad una macchina, all'inizio magari solo un calcolo, ma poi, chissà dove si poteva arrivare. Il computer era come un bambino buono ed intelligente, con molte potenzialità, ma a cui bisognava dire tutto; ed affascinante era anche il modo in cui bisognava parlare al computer: senza ambiguità, seguendo dei passi precisi, completamente specificati. Per chi, come me, non era abituato a questa filosofia "algoritmica" si apriva un mondo nuovo di idee e di possibilità.

Ecco, queste furono le mie emozioni nell'entrare nel mondo dei computer, il fascino irresistibile che subii. Oggi, che i computer iniziano ad essere alla portata di tutti, questo fascino colpisce ancora, più di allora. Lo vedo appunto dalle lettere dei miei lettori: gli studenti, i professionisti, gli hobbysti.

Personalmente, penso che la spiegazione di tutto ciò sia semplice. Un impianto Hi-Fi, per quanto complesso e perfezionato, è uno strumento che riesce a fare una sola cosa: riprodurre musica. Una macchina fotografica, una lavatrice, un aeroplano, sono macchine meravigliose, ma possono fare solo quello per cui sono state progettate. Un computer, da solo, non può fare nulla, ma, paradossalmente, è questo il suo grande pregio: è una macchina universale, che funziona secondo gli ordini che le si danno. Se non sbaglio, è la prima volta nella storia che l'uomo ha una simile possibilità. E ciò naturalmente non può lasciare indifferenti.

Questo numero è dedicato ai pocket computer. Si tratta dell'aspetto più moderno dell'idea di computer: in essi il fascino dell'hardware e del software si unisce alla possibilità di fare grandi cose con piccoli mezzi; ed inoltre sono in gran parte responsabili dello sviluppo dell'informatica personale. I calcolatori tascabili programmabili e i pocket computer hanno fatto provare a milioni di persone il piacere di programmare un computer, la gioia di dare anima a una macchina inanimata. Beh, sarete d'accordo con me, non è poco.

Nel prossimo numero

Un data base modulare per
l'Apple

Le variabili di sistema nello ZX81

Come stampare lo schermo del
VIC

I segreti dei personal: TI99/4A,
Nuova Elettronica, Apple,
PET/CBM, VIC 20, ZX81

Raccolta di routine Basic: la fase
della luna

Un piccolo parallelo tra Pocket Basic

di Paolo Capobussi

Un confronto tra le possibilità dei tascabili che adottano un interprete per il diffusissimo linguaggio Basic non è mai un compito semplice. Neppure può portare a risultati soddisfacentemente esaurienti.

A questo livello, infatti, la capacità di memoria generalmente è talmente ridotta che piccole differenze possono talvolta portare a grandi vantaggi, o svantaggi, nell'uso efficiente di questi minimi elaboratori.

Nella pratica non è solo il linguaggio adottato e la sua estensione che influiscono a livello di facilità e agiatezza operativa.

Sono quindi lecite diverse domande. È più comodo il milione e mezzo (quasi) richiesto per acquistare la HP-75C con una tastiera normale e mille funzioni richiamabili tramite particolari sequenze di tasti, oppure il mezzo milione (quasi e non solo: stampante e interfaccia compresi) richiesto per l'acquisto della Casio FX-702P, che ha una tastierina piccola, ma completa di tutte le funzioni possibili e con addirittura tastierino numerico separato e sempre pronto all'uso?

Non si può rispondere per reazione "epidermica", dettata dalla simpatia o antipatia per quel prodotto.

Come possiamo notare dalle tabelle riportate, infatti, l'estensione dell'interprete Basic e quindi le funzioni supportate dal pocket sono notevolmente differenti. Dipende tutto, come sempre, dall'uso che se ne vuole fare.

FUNZIONI DEL BASIC

	HP-75C	PC-1500	FX-702P	PB-100	PC-1211 TRS 80P	PC-1251
1 ABS	*	*	*	*	*	*
2 ACOS	*	*	*	*	*	*
3 ANGLE	*	-	-	-	-	-
4 ASIN	*	*	*	*	*	*
5 ATN	*	*	*	*	*	*
6 CAT \$	*	-	-	-	-	-
7 CEIL	*	-	-	-	-	-
8 CMR \$	*	*	-	*	*	*
9 COS	*	*	*	*	*	*
10 COT	*	-	*	*	-	-
11 CSC	*	-	*	*	-	-
12 DATE	*	-	-	-	-	-
13 DATE \$	*	-	-	-	-	-
14 DEG	*	*	*	*	*	*
15 EPS	*	-	-	-	-	-
16 ERRL	*	-	-	-	-	-
17 ERRN	*	-	-	-	-	-
18 EXP	*	*	*	*	*	*
19 FLOOR	*	-	-	-	-	-
20 FP	*	-	FRAC	-	-	-
21 INF	*	-	-	-	-	-
22 INT	*	-	-	-	-	-
23 IP	*	INT	INT	INT	INT	INT
24 KEY \$	*	*	*	*	*	*
25 LEN	*	*	*	*	*	*
26 LOG	*	LN	LN	LN	LN	LN
27 LOG 10	*	*	*	*	*	*
28 MAX	*	-	-	-	-	-
29 MEM	*	*	DEFM	DEFM	*	*
30 MIN	*	-	-	-	-	-
31 MOD	*	-	-	-	-	-
32 NUM	*	*	*	*	*	*
33 PI	*	*	*	*	*	*
34 POS	*	-	-	-	-	-
35 RAD	*	*	*	*	*	*
36 RES	*	-	ANS	-	-	-
36 RMD	*	-	-	-	-	-
38 RND	*	*	*	*	*	*
39 SEC	*	-	-	-	*	-
40 SGN	*	*	*	*	*	*
41 SIN	*	*	*	*	*	*
42 SQR	*	*	*	*	-	*
43 STR \$	*	*	-	-	-	*
44 TAB	*	*	CSR	CSR	*	-
45 TAN	*	*	*	*	-	*
46 TIME	*	*	-	-	-	-
47 TIME \$	*	-	-	-	-	-
48 UPRC \$	*	-	-	-	-	-
49 VAL	*	*	-	*	-	*
50 VER \$	*	-	-	-	-	-

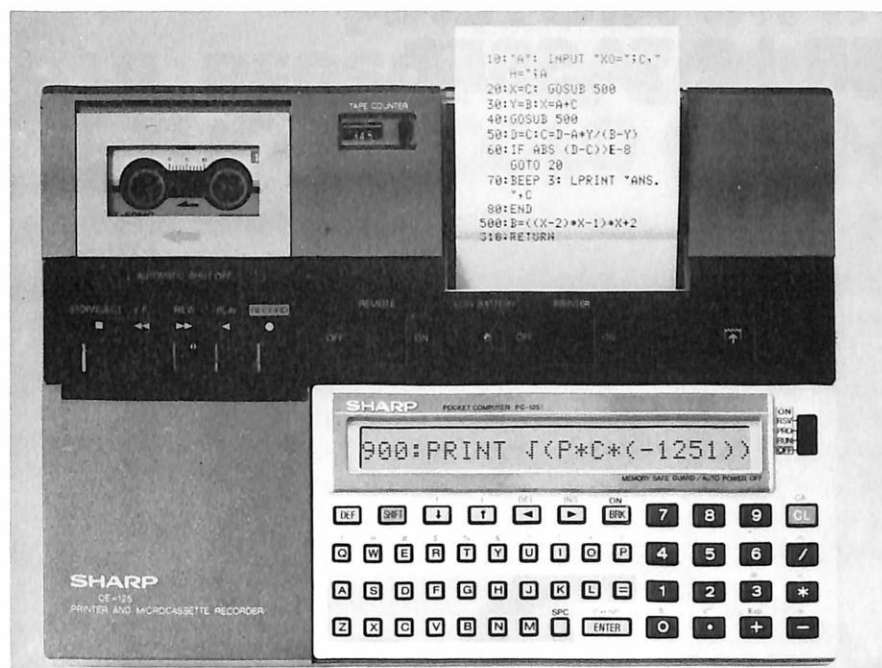


Figura 1. Il compattissimo PC-1251 della Sharp, un pocket in Basic che racchiude in sé molte potenti funzioni. È realmente ciò che presagivano lo scorso novembre sul fascicolo speciale di BIT: il Basic credit-card. Inoltre permette di tenere in borsa anche il completo stampante-registratore a microcassette, veramente un'ottima soluzione per rendere velocemente utilizzabili i pocket, altrimenti sottoimpiegati.

civile, ove pare che fin dall'epoca delle passate calcolatrici tascabili il mercato dei pocket abbia dato una spinta decisiva alla rapidità di calcolo?

Tutti problemi che il compratore dovrà "attentamente valutare", come ormai è abitudine dire quando le variabili in gioco sono così tante che i gusti personali prendono spesso il sopravvento.

Comunque, proprio per evitare queste reazioni totalmente istintive, a volte comunque decisive e non sempre trascurabili, pubblichiamo i confronti detti tra le funzioni, gli statement, i comandi dei pocket in Basic più potenti e diffusi.

Il termine di confronto possiamo ritenere essere l'elaboratore tascabile della HP, un gioiello non troppo piccolo e neppure troppo poco costoso, ma certamente di potenza e qualità ineccepibili.

Quest'ultimo è dato con un interprete Basic che rispetta pienamente le caratteristiche codificate del Minimal Basic ANSI, secondo le specifiche depositate al NBS (National Bureau of Standards). Piccole differenze sono ampiamente documentate sul manuale, e comunque sono veramente minime e intuitive.

Inoltre comprende una serie di comandi e di capacità operative che hanno una versatilità imprevedibile, nel senso che sarebbe veramente impensabile riuscire a prevedere tutte le possibili applicazioni. Solamente la potentissima gestione di stringhe e variabili legate a contatori di tempo, la capacità di pilotare catene HP-Interface Loop acquisendo dati dalla strumentazione più diversa, la ridefinibilità della tastiera, danno una pallida idea del per-

OPERATORI ARITMETICI E LOGICI					
HP-75C	PC-1500	FX-703P	PB-100	PC-1211 TRS 80P	PC-1251
+	+	+	+	+	+
-	-	-	-	-	-
*	*	*	*	*	*
/	/	/	/	/	/
^	AND			AND	AND
DIVor	OR			OR	OR
AND	NOT				NOT
OR	EXOR				
NOT					

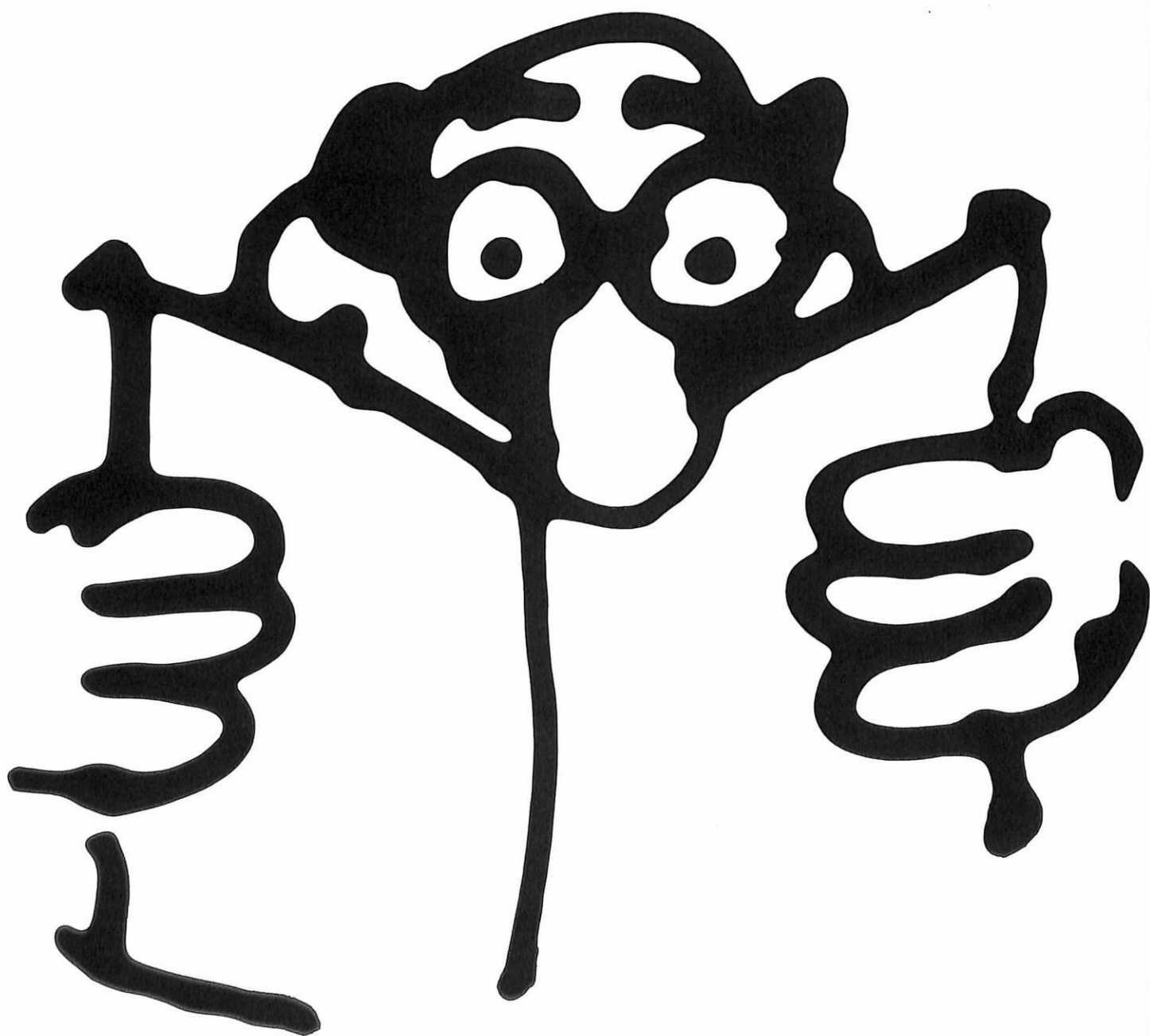
OPERATORI RELAZIONALI						
	HP-75C	PC-1500	FX-702P	PB-100	PC-1211 TRS 80P	PC-1251
=	Idem	Idem	Idem	Idem	Idem	
<						
>						
or						
#						
<=						
>=						
<>						
=						

COMANDI DI MODO TIME NON PROGRAMMABILI
HP75C
ADJUST
EXACT
EXTO
RESET
SET
STATS

Un ulteriore mezzo di scelta è la diffusione del software disponibile. È facilmente raggiungibile, come quello pubblicato su riviste a larga

diffusione o ottenibile tramite club? Oppure è limitato a poche applicazioni, a pochi campi specialistici quale il tipico settore dell'ingegneria

**E CHI MI AIUTERA'
PER LE TASSE?**



IL PERSONAL COMPUTER IBM IL TUO PICCOLO GRANDE AMICO.

Ore e ore passate sui libri contabili e registri IVA: ecco cosa vogliono dire le tasse. Ora puoi tirare un sospiro di sollievo; perchè c'è un amico pronto a darti una mano: il Personal Computer IBM.

Così piccolo da stare sulla tua scrivania, tanto grande da aiutarti a risolvere tutti i tuoi problemi di tasse. E non solo quelli. Perchè il Personal Computer IBM può fare di tutto: riceve dati, calcola, fa statistiche, registra, controlla, archivia e

stampa. E grazie alla sua potente memoria ti consente di gestire un'infinità di informazioni.

Non è necessario essere un addetto ai lavori per imparare ad usarlo.

Vedrai, in poche ore diventerete ottimi amici. Perchè ragiona come te. Vuoi metterlo alla prova? Vai da un concessionario IBM per il Personal Computer IBM. Il tuo piccolo grande amico ti sta aspettando.



IBM Italia
Distribuzione Prodotti srl



Il Personal Computer IBM contiene un microprocessore a 16 bit e una memoria di utilizzo che raggiunge i 640 Kbyte. E, grazie ai dischi fissi, la capacità massima di memoria del sistema è di 21 Mbyte in linea. Inoltre, puoi facilmente collegarti con un altro Personal Computer IBM, con elaboratori più potenti e con la rete dei Centri Servizi Elaborazione Dati della IBM.

Sistemi operativi: DOS 1-DOS 2-UCSD-CP/M-86. **Supporti per le comunicazioni:** Supporto per Comunicazioni Asincrone - Supporto per Comunicazioni SDLC - Programma di Emulazione 3101 - Programma di Emulazione 3270.

Programmi applicativi: Corso Autodidattico Interattivo - Gestione Aziendale - EasyWriter (dal 20/5 anche in italiano) - Multiplan (dall'8/6 anche in italiano) - VisiCalc.

COMANDI DI SISTEMA					
HP-75C	PC-1500	FX-702P	PB-100	PC-1221 TRS 80P	PC-1251
ALARM OFF ALARM ON ASSIGN IO AUTO BEEP PFF BEEP ON BYE CAT CAT ALL CAT CARD CLEAR LOOP CLEAR VARS CONT COPY DEFAULT OFF DEFAULT ON DEF KEY DELAY DELETE DISPLAY IS EDIT ENDILE FETCH FETCH KEY INITIALIZE LIST LIST IO LOCK MARGIN MERGE NAME OFF IO OPTION ANGLE DEGREES OPTION ANGLE RADIAN PACK PLIST PRINTER IS PROTECT PURGE PWIDTH RENAME... TO RENUMBER RESTORE IO RUN STANDBY OFF STANDBY ON TRACE FLOW TRACE OFF TRACE VARS TRANSFORM UNPROTECT WIDTH	COLOR CONT CSIZE GLCOURSOR GRAPH LCURSORS LIST LF LINE LLIST NEW NEWO RUN RLINE ROTATE LPRINT STATUS DEBUG TET MEM CHAIN CLOAD CLOAD? CSAVE IMPUT= MERGE PRINT= RMT OFF RMT ON	RUN CLR CLR ALL LIST LIST V LIST ALL DEFM PASS SAVE SAVE ALL LOAD LOAD ALL VER STAT DEL SAC	RUN CLEAR CLEAR A LIST LIST A EXT DEFM SAVE SAVE A LOAD LOAD A VER	RUN DEBUG CONT LIST NEW CSAVE CLOAD CLOAD? CHAIN PRINT # INPUT Ø	RUN NEW LLIST LPRINT PASS LIST CONT TR ON TR OFF CSAVE CLOAD CLOAD? CHAIN MERGE PRINT # INPUT #

ché siano così completi ed estesi gli elenchi delle possibili funzioni documentate sull'ottimo manuale, un vero piccolo capolavoro della manualistica di casa Packardiana.

Un altro motivo che porta a scegliere HP come attuale riferimento per un corretto confronto è il fatto che tutte le altre macchine scendono spesso a compromessi. Tipico è l'esempio dell'istruzione DIM. In equivalente su FX-702p è solo un equivalente apparente, intuitivo. Si può infatti riconoscere una analoga istruzione DEFM, che riserva dello

spazio per le variabili, a scapito della zona di programmazione. È un comando che però va dato solo a livello di tastiera, come vero e proprio comando di sistema. Non è possibile introdurlo a programma, ovvero non è possibile, da programma, una definizione dell'occupazione spazio di memoria riservata a variabili.

È, allora, confrontabile con il tipico DIM presente nel vero ANSI Basic? Ad una prima occhiata riteniamo di sì, infatti è in tale situazione che l'abbiamo paragonato. Certa-

Figura 2. Un confronto tra Basic presenti attualmente sui pocket computer più diffusi.

Alla completezza dello standard ANSI adottato da HP, fanno ottimo riscontro anche gli altri pocket, seppure con capacità ridotte, come d'altronde sono ridotte le dimensioni e... il prezzo.

In realtà il confronto diretto è molto difficile, anche e soprattutto per le funzioni di sistema, caratteristiche della struttura di ogni singola macchina. Le tabelle, comunque, danno una non vaga idea delle possibilità di questi tascabili.

mente ciò attirerà comunque le ire dei puristi, e farà gridare allo scandalo i più o meno esperti del campo.

Ma come potremo altrimenti effettuare un confronto? Quale l'istruzione che più si avvicina, rispettando una esigenza di capire quali sono i buchi, o banchi, del sistema?

Un poco alla rinfusa, unico modo che ci sembra possibile per non ripetere pedissequamente delle impressioni che possono tranquillamente derivare dalla lettura delle tabelle riportate, accenniamo a qualche particolarità delle singole macchine.

Immediatamente si ha l'impressione che la piccolissima PC-1251, il pocket che di per sé sta veramente nel taschino della giacca, sia di una completezza strabiliante. Se non fosse per l'assenza delle opzioni grafiche potremmo dire che ci avvicina di più ad un Basic quale quello della possente PC-1500 che, come tutti hanno potuto apprezzare, è stato il primo passo verso il vero personal-da-tasca. L'equivalente Casio, purtroppo, non esiste con le stesse caratteristiche di massima portatilità e comodità d'uso delle periferiche. Se infatti è vero che possiamo tutti apprezzare le ridotte dimensioni e il basso costo del tascabile per studenti PB-100, è altrettanto vero che grande è la comodità di potersi portare appresso un registratore e la stampante senza alcun problema di collegamento, alimentazione separata, incompatibilità frequente tra registratori di marche molto diverse e analoghe "scocciature".

ECCO CHI TI AIUTERÀ AD ANDARE D'AMORE E D'ACCORDO CON IL TUO NUOVO AMICO.



Il tuo concessionario IBM.

Ti aiuterà a ottenere il massimo dal tuo Personal Computer IBM. Ti garantirà un'assistenza puntuale e un servizio all'altezza del nome IBM, che in tutto il mondo significa efficienza e affidabilità. Per una lunga e proficua amicizia fra te e il tuo Personal Computer IBM.

Aosta
INFORMATIQUE SAS - Av. Du Cons. Des Commis, 16 -
11100 Aosta - Tel. 0165.2242

Bari
PASED SRL - Via Calefati, 134/136 - 70125 Bari -
Tel. 080/481488-481641

Belluno
SCP COMPUTER SYSTEM SRL - Via Feltre, 32 -
32100 Belluno - Tel. 0437.20826

Bergamo
NUOVA INFORMATICA SAS - Via Provinciale, 86 -
24021 Albino - Tel. 035.751784
SELTERING SPA - Via Verdi, 25 - 24100 Bergamo -
Tel. 035.248256/7/8

Bologna
ABACO INFORMATICA SAS - Via Bernini, 1 -
40138 Bologna - Tel. 051.393274
PALAZZO DONATO - Via Emilia, 23/A - 40026 Imola
Tel. 0542.29195
SYSDATA ITALIA SPA - Via M. D'Azeglio, 58 -
40123 Bologna - Tel. 051.330021

Brescia
FIN-ECO SERVICE SRL - Via G. Rosa, 34 -
25100 Brescia - Tel. 030.59055
MICROSELT SRL - Via Cipro, 33 - 25125 Brescia -
Tel. 030.220391/220513
SELTERING SPA - Via Cipro, 33 - 25125 Brescia -
Tel. 030.220391

Campobasso
PUBBLISISTEMI SRL - Via S. Antonio Abate, 231 -
86100 Campobasso - Tel. 0874.98141

Como
BRUNO SRL - Via Rubini, 5 - 22100 Como - Tel. 031.260538
ZECCA INFORMATICA SPA - Viale Dante, 14 -
22053 Lecco - Tel. 034.373290

Cosenza
CALIO SRL - Via N. Serra, 90 - 87100 Cosenza - Tel. 0984.32807

Cuneo
SISTEMI SRL - Via Giolitti, 26 - 12100 Cuneo -
Tel. 0171.55475/6

Firenze
C.C.S. SAS - Viale Repubblica, 298 - 50047 Prato -
Tel. 0574.580222/593865
SAL DISTRIBUZIONE SRL - Punto Vendita SESA -
Via delle Panche, 65 - 50100 Firenze - Tel. 055.411635
SESA DISTRIBUZIONE SRL - Via XI Febbraio, 24 B
50053 Empoli - Tel. 0571.72148/75196

Forlì
HARD & SOFT SYSTEMS SRL - Via Valturio, 43 -
47037 Rimini - Tel. 0541.773343

Frosinone
SAIU ELETTRONICA SRL - Via Vado del Tufo, 85 -
03100 Frosinone - Tel. 0775.83093

Genova
DIFFEL SRL - Via XX Settembre, 31/4 - 16121 Genova
Tel. 010.592431/595669

Lecce
S.V.I.C. SRL - Via V. Emanuele, 121 - 73024 Maglie -
Tel. 0836.21604

Messina
SICIL FORNITURE SPA - Via Don Blasco, 75 -
98100 Messina - Tel. 090.2923987

Milano
ELEDRA 3S SPA - Viale Elvezia, 18 - 20154 Milano -
Tel. 02.349751
EDICONSULT SRL - Via Rosmini, 3 - 20052 Monza -
Tel. 039.389850
HOMIC PERSONAL COMPUTER SRL
Piazza De Angeli, 3 - 20146 Milano - Tel. 02.4988201
HUGNOT LUIGI LUCIANO - Via De Togni, 10 -
20123 Milano - Tel. 02.873190/865641
MICROTECH SRL - Via F.lli Bronzetti, 20 -
20129 Milano - Tel. 02.733609/740654
S.D.I. INFORMATICA SPA - Via G. Winckelmann, 1
20146 Milano - Tel. 02.470846
SIRIO SHOP SRL - Viale Certosa, 148 - 20156 Milano
Tel. 02.3010051/305778
SOFTEC SRL - Viale Maino, 10 - 20129 Milano -
Tel. 02.7491196
TRANSDATA SRL - Milano Fiori Palazzo E3 Str 1ª -
20094 Assago - Tel. 02.8242460/8251066

Napoli
POINTER SRL - Via A. De Gasperi, 45 - 80133 Napoli
Tel. 081.312312

Padova
CERVED ENGINEERING SPA - C.so Stati Uniti, 14 -
35100 Padova - Tel. 049.760733

Palermo
TESI SRL - Via E. Notarbartolo, 23 - 90141 Palermo -
Tel. 091.260549

Pavia
I.T.C. INFORMATICA SRL - Strada Nuova, 86 -
27100 Pavia - Tel. 0382.303201
LOGICA INFORMATICA SRL - Via Montegrappa, 32
27029 Vigevano - Tel. 0381.81888

Perugia
PUCCIUFFICIO SNC - Via XX Settembre, 148C -
06100 Perugia - Tel. 075.72992/74910

Roma
CERVED SPA - Via Appia Nuova, 696 - 00100 Roma
Tel. 06.7940241
DATAOFFICE SPA - Via Sicilia, 205 - 00187 Roma -
Tel. 06.4754568/464010

ELEDRA 3S SPA - Via G. Valmarana, 63 - 00100 Roma -
Tel. 06.8127324/8127290
GEDIN SRL - L.go D. De Dominicis, 7 - 00159 Roma -
Tel. 06.432183/4389360
I.S.E.D. SPA - Via Tiburtina, Km. 12,300 - 00131 Roma
Tel. 06.4125851
JACOROSI SPA - Via V. Brancati, 64 - 00144 Roma
Tel. 06.54916
SAPES SRL - V.le Tito Livio, 12 - 00136 Roma -
Tel. 06.3453536/3452197
VALDE ADEL SRL - Piazza S. Anastasia, 3 - 00100 Roma -
Tel. 06.6786663

Salerno
OMNIA SRL - C.so Garibaldi, 47 - 84100 Salerno -
Tel. 089.353914/220366

Siena
SILOG SISTEMI LOGICI SRL - Via Sicilia, 5 -
54085 Monteriggioni - Tel. 0577.54085

Terni
DPS SRL - Via Pacinotti, 6 - 05100 Terni - Tel. 0744.58247

Torino
DIVERSIFICATE VENCO SRL - C.so Matteotti, 32A -
10121 Torino - Tel. 011.545525/545125
SISTEMI SPA - C.so Peschiera, 240 - 10139 Torino -
Tel. 011.3358676
SOFTEC SRL - C.so San Maurizio, 79 - 10124 Torino -
Tel. 011.8396444

Treviso
EDS SRL - Via S. Pio X, 154 - 31033 Castelfranco Veneto
Tel. 0423.490178
INFORMATICA TRE SRL - Viale della Repubblica, 19
31100 Treviso - Tel. 0422.65993

Trieste
DITTA MURRI - Via A. Diaz, 24/A - 34123 Trieste -
Tel. 040.733253/734383

Varese
ELMEC SPA - Via Sebenico, 12 - 21100 Varese -
Tel. 0332.264135

Venezia
COMPUTIME SRL - Piazza Rizzo, 63 -
30027 S. Donà di Piave - Tel. 0421.2548

Vercelli
ANALOG SNC - Via Dionisotti, 18 - 13100 Vercelli -
Tel. 0161.61105

Verona
PRAGMA SERVICE SRL - Via Carmelitani Scalzi, 20 -
37100 Verona - Tel. 045.24629/38522

Viterbo
ITALBYTE SRL - V.le Trento-Pal. Garbini -
01100 Viterbo - Tel. 0761.221333

● E per acquisti superiori alle 20 unità puoi anche rivolgerti alle filiali IBM.

● Per ulteriori informazioni sugli indirizzi dei punti di vendita telefona a 02/21752360 oppure 06/54864962.

IBM

IBM Italia
Distribuzione Prodotti srl

STATEMENT DEL BASIC						
	HP-75C	PC-1500	FX-702P	PB-100	PC-1211 TRS 80P	PC-1251
1	ASSIGN	*	-	-	-	-
2	BEEP	*	*	-	-	*
3	CALL	*	-	GSB #	GOSUB #	-
4	DATA	*	*	-	-	*
5	DEF FN	*	-	-	-	-
6	DIM	*	*PRINT	DEFM	DEFM	-
7	DISP	*	PRINT USING	PRT	-	-
8	DISP USING	*	*	PRT USING	-	PRINT
9	END	*	-	*	*	PRINT USING
10	END DEF	*	*	-	-	*
11	FOR...TO...STEP	*	*	*	*	*
12	GOBUS	*	*	*	*	*
13	GOTO	*	IF...THEN	*	*	*
14	IF...THEN...ELSE	*	-	IF...THEN	IF...THEN	INF...THE,
15	IMAGE	*	*	-	-	-
16	INPUT	*	-	*	*	*
17	INTEGER	*	*	-	-	-
18	LET	*	-	-	-	*
19	LET FN	*	*	-	-	*
20	NEXT	*	-	*	*	*
21	OF ERROR	*	-	-	-	-
22	OFF TIMER #	*	-	-	-	-
23	ON ERROR	*	-	-	-	-
24	ON TIMER #	*	*	-	-	-
25	ON...GOBUS	*	*	GSB(n)	GOSUB(n)	-
26	ON...GOTO	*	-	GTO(n)	GOTO(n)	*
27	OPTION BASE	*	-	-	-	-
28	POP	*	LRPINT	-	-	-
29	PRINT	*	-	MODE7	*	*
30	PRINT #	*	-	-	-	-
31	PRINT USING	*	-	*	*	*
32	PUT	*	*	-	-	-
33	RANDOMIZE	*	*	-	-	-
34	READ	*	-	-	-	-
35	READ #	*	-	-	-	-
36	REAL	*	*	-	-	-
37	REM	*	*	-	-	*
38	RESTORE	*	-	-	-	-
39	RESTORE #	*	*	-	-	*
40	RETURN	*	-	*	*	-
41	SHORT	*	*	-	-	-
42	STOP	*	*	*	*	*
43	WAIT	*	*	*	-	*

L'equivalente effettivo la Casio, in realtà, l'aveva in cantiere: era il FX-801P. Ma questo pocket, pur rispettando la compattezza e la completezza di una versione integrata con stampante e registratore della diffusa FX-702P, ha evidentemente risentito di un mercato in rapidissima evoluzione, che ha visto la serie FX rapidamente perdere una parte del brillante smalto che possedeva all'inizio.

In Francia, paese ove la commercializzazione era iniziata, sembra che il prodotto stia scomparendo. L'unico modo per correre ai ripari sarebbe, a nostro parere, una macchina con capacità circa doppie della FX-702P, ma con dimensioni da PC-1251, integrata naturalmente a registratore a microcassette e stampante

compresa, speriamo ancora su carta metallizzata.

Una macchina di tali caratteristiche diverrebbe allora veramente appetibile per il mercato infinito degli uomini di affari, dei professionisti, degli ingegneri. Se venissero mantenute le caratteristiche scientifiche e matematico-statistiche della FX-702P nella loro interezza, corredando il tutto di biblioteche di microcassette per la soluzione dei problemi più diversi che non siano i soliti giochi o le conversioni di base spesso malfatte, allora, crediamo, ci sarebbe una effettiva esplosione dell'uso di tali pocket. In un mercato medio-alto, se le dimensioni rimanesse ridotte al livello detto, certamente si avrebbe una parziale sostituzione delle normali calcolatrici

quattro operazioni con i più versatili pocket. Sarebbe una gara a possedere sulla propria scrivania una calcolatrice semplice da usare, con stampante, portatile all'occorrenza e con capacità di soluzione di piccoli problemi complessi al premere di un tasto.

Se le capacità di memorizzare saranno adeguatamente sviluppate non è impensabile che consulenti o software-house possano produrre, per i distributori stessi del prodotto, microcassette di programmi, gestionali e no, già pronti e registrati, capaci di risolvere molti problemi comuni al nostro mercato.

In Inghilterra, per la FX-702P, viene addirittura commercializzata una cassetta che è definibile un VisiCalc in miniatura. Ovvero un pro-

gramma di planning dalle infinite capacità di correlazione dei dati introdotti.

Un nome manca all'appello: Texas Instruments. Da anni impegnata nel settore programmabili, dopo aver riscosso notevoli successi con la filosofia del software a stato solido, sembra impegnata in un ripensamento di strategia di notevole portata.

Due colpi sono però stati annunciati anche in Italia, e speriamo che possano presto essere messi a segno: il professional computer, che per ora non riguarda questa sede di discussione, e il CC-40, che invece andrebbe paragonato al HP-75C. Con quest'ultimo, di fattezze notevolmente interessanti, ma non ancora valutabile per la materiale assenza fisica che ci impedisce di inserirlo nelle tabelle dei confronti, la Texas si è finalmente buttata nella mischia dei portatili in Basic. Parte già completo di interfaccia standard RS-232 e varie periferiche tra cui delle cassette a nastro particolarmente compatte e sottili. Quindi, è prevedibile, con un software di gestione del tutto all'altezza, almeno, della diretta concorrente HP. Ma, in fondo, perché sempre giocare sui confronti? A questo livello pare proprio che l'originalità accompagni tutti, e la eventuale scelta di acquisto o il puro tifo dettato dalla passione per questi aggeggi, debba essere più una questione personale.

In fondo sono strumenti, e nessuno strumento, per quanto perfetto sia, serve a qualcosa se dietro non c'è un uomo che lo usa.

VIDEO GIOCHI

LA PRIMA E UNICA
RIVISTA DI VIDEOGAMES
COMPUTER
GIOCHI ELETTRONICI



Una pubblicazione
del Gruppo Editoriale Jackson

è in edicola il nuovo numero



RISERVATO PERSONAL
64 pagine di programmi
per il tuo Personal Computer

BITEST CANON AS 100

DBASE II
TUTTO SULL'IEEE 488
IL LINGUAGGIO FORTH
FLOPPY DISK CONTROLLER



UNA PUBBLICAZIONE DEL
GRUPPO EDITORIALE JACKSON

Assolutizzare le etichette

di Paolo Capobussi

Il problema tipico dei programmatori di TI-59, la calcolatrice Texas Instruments che ancora fa parlare di sé, è sempre stato quello del passaggio da un listato concepito con etichette a quello con una serie di indirizzi assoluti.

La Texas TI-59, infatti, nonostante sia stata introdotta nel mercato nel lontano 1977, quando ancora i pocket computer erano considerati giocattolini per principianti, permette di porre all'inizio di ciascuna routine una etichetta che potremmo definire "alfanumerica". È così possibile far seguire al codice del tasto Lbl il codice di un altro tasto qualsiasi, che verrà considerato non più solvente la sua propria funzione, ma quale indirizzo richiamabile di inizio routine.

Sarà quindi successivamente possibile eseguire dei branch a tale etichetta tramite qualunque delle istruzioni di salto previste.

È evidente che un tale metodo, non previsto neppure da linguaggi potenti come il Basic di molti personal, porta a delle notevoli facilitazioni soprattutto nella stesura dei programmi. Si pensi la comodità di non doversi più preoccupare degli indirizzamenti. Un GOTO Lbl rimane tale anche dopo qualunque correzione abbia portato a variazioni del numero di passi di programma, ovvero della posizione delle singole istruzioni che il GOTO stesso richiama. Come per tutte le comodità concesse da queste macchinette esiste, come sempre, il cosiddetto rovescio della medaglia. L'uso di tale tecnica spreca un passo in più

rispetto a quanti ne verrebbero consumati utilizzando gli indirizzi assoluti. Ciò appare quantomeno ridicolo a chi è abituato a ben altre capacità di memoria: in fondo un passo solo per etichetta non dovrebbe costituire un problema. Nulla di più falso! Il buon programmatore di TI 59 è sempre angustiato dall'incertezza di non riuscire a inserire tutto il programma per problemi di quantità di memoria. In queste condizioni, date dalla limitazione massima dei 960 passi programma, ogni suggerimento per impaccare le proprie istruzioni è, da sempre, molto ben accolto.

Non è solo il problema di spazio, comunque, l'unico reale inconveniente che tocca la problematica dell'uso di etichette. Se il programma è piuttosto lungo o impiega diverso tempo per portare a dei risultati, trasformare tutte le etichette e i relativi richiami in codici di indirizzo assoluti diventa necessità impellente.

Un esempio tipico è un programma ricorsivo che debba stampare grandi quantità di dati utilizzando sempre le stesse routine. L'uso di etichette limita la velocità in quanto la calcolatrice inizia la ricerca della Label interessata sempre da passo 000, ovvero dall'inizio della memoria programma. Il tempo di ricerca è talvolta tanto elevato che l'eliminazione di etichette con indirizzi assoluti porta a risparmi addirittura valutabili attorno al 40% del tempo totale di svolgimento.

Purtroppo non esiste una funzione specifica del monitor di sistema

che permetta l'eliminazione automatica delle etichette e il loro rimpiazzo con indirizzi assoluti. Peraltro sarebbe forse pretendere troppo.

Sembra, infatti, un problema banale, ma è invece un lavoro che richiede particolare attenzione e continue modifiche, dovute al fatto che, via via che si prosegue nella trasformazione, altri indirizzi già supposti esatti vengono modificati, con evidente confusione e perdita di tempo. Si rende allora necessario un metodo razionale e preciso che eviti ogni possibile errore e con poco sforzo porti ad un risultato corretto.

Il metodo qui proposto è, naturalmente, manuale, nel senso più vero della parola. È escluso, infatti, poterlo eseguire con l'ausilio della macchina stessa. Vuole quindi essere una indicazione di un abile sistema di procedere nella definitiva stesura del proprio parto di programmazione.

La prima operazione da eseguire è stampare un listato completo del programma etichettato. I possessori di stampante potranno ottenerlo con la semplice operazione 2nd ist. In tal modo la PC100, ovvero la stampante associabile alla TI 59, produrrà una striscia contenente tutte le indicazioni a noi necessarie.

Coloro i quali non possiedono la stampante dovranno, con certissima pazienza, scrivere un listato equivalente su un semplice foglio di carta. La seconda operazione consiste nel prendere nota del nome di tutte le etichette utilizzate ed il passo a cui effettivamente inizia la routine etichettata. Pertanto supponendo di

	000	76	LBL	
	001	11	A	
	002	42	STD	
	003	01	01	
	004	61	GTO	
	005	66	PAU	+1→
	006	76	LBL	
	007	42	STD	
-2→	008	05	5	
	009	00	0	
	010	42	STD	
	011	00	00	
	012	76	LBL	
	013	66	PAU	
-2→	014	85	+	
	015	01	1	
	016	95	=	
	017	66	PAU	
	018	97	DSZ	
	019	00	00	
	020	66	PAU	+1→
	021	61	GTO	
	022	42	STD	+1→
	023	00	0	
	001	11	A	
001	~~007~~	42	STD	
014	~~013~~	66	PAU	

Figura 1. La stampa del listato mostrata in questa figura riporta la situazione iniziale, di preparazione, per ridurre tutte le etichette ad indirizzi assoluti.

	000	76	LBL	
	001	11	A	
	002	42	STD	
	003	01	01	
	004	61	GTO	
	005	66	PAU	+1→
	006	76	LBL	
	007	42	STD	
-2→	008	05	5	TOT.-1
	009	00	0	
	010	42	STD	
	011	00	00	
	012	76	LBL	
	013	66	PAU	
-2→	014	85	+	TOT.-3
	015	01	1	
	016	95	=	
	017	66	PAU	
	018	97	DSZ	
	019	00	00	
	020	66	PAU	+1→
	021	61	GTO	
	022	42	STD	+1→
	023	00	0	
	001	11	A	
007	~~007~~	42	STD	
014	~~013~~	66	PAU	

Figura 2. Dopo aver eseguito i semplici conti indicati nell'articolo, ecco la situazione che si ottiene sul listato. A questo punto si possono eliminare tutte le etichette, iniziando così la trasformazione di tutti gli indirizzi.

	000	76	LBL	
	001	11	A	
	002	42	STD	
	003	01	01	
	004	61	GTO	
	005	00	00	
	006	11	11	
	007	05	5	
	008	00	0	
	009	42	STD	
	010	00	00	
	011	85	+	
	012	01	1	
	013	95	=	
	014	66	PAU	
	015	97	DSZ	
	016	00	00	
	017	00	00	
	018	11	11	
	019	61	GTO	
	020	00	00	
	021	07	07	
	022	00	0	
	023	00	0	

Figura 3. Il listato ottenuto con la PC-100, come si presenta dopo aver effettuato la "pulizia" dalle etichette.

avere la sequenza di codici mostrata in fig. 1, annoteremo che al passo 017 - reale inizio della routine - corrisponde l'etichetta RCL.

Chi possiede la stampante PC100 potrà ancora avvalersi di una "facilitie" di tale periferica. Sarà infatti semplice premere in successione i tasti 2nd Op 08 per ottenere l'elenco di tutte le etichette utilizzate dal programma. È però necessario por-

re attenzione che così facendo gli indirizzi stampati in corrispondenza a ciascuna etichetta non sono, in realtà, gli indirizzi di inizio routine. Sarà quindi necessario sommare a tutti questi indirizzi stampati una unità.

Il terzo passo di questo procedimento richiede di indicare a sinistra del listato una freccetta rivolta verso l'interno per ogni indirizzo di inizio

routine. A ciascuna di queste frecce si attribuisce il numero -2.

Poi si tracceranno altre freccette, questa volta a destra e rivolte verso l'esterno con il numero +1, in corrispondenza di tutti i richiami a tali etichette, siano essi istruzioni come GOTO, SBR, Dsz o qualsiasi altra. La fig. 2 mostra un listato dopo aver eseguito le indicazioni date fino ad ora.

Infine, iniziando da passo 000, si eseguirà la somma algebrica dei +1 e dei -2 scritti sopra le freccette, fino a raggiungere l'inizio di una routine etichettata, ovvero fino alla prima freccetta con scritto -2 (compresa). In corrispondenza ad essa si scrive il totale.

La somma algebrica del numero ottenuto, con il numero costituente l'indirizzo corrispondente a quella etichetta e già segnato nell'apposito elenco che era stato costruito inizial-

mente, costituisce il vero indirizzo assoluto di tale etichetta.

Ripetendo tale procedura fino al termine del programma, ovvero fino a che non ci siano più etichette, si determinano tutti gli indirizzi assoluti richiesti.

A tal punto è banale una modifica del programma caricato in memoria, facendo accorto uso degli appositi tasti Ins e Del. Questi permetteranno di cancellare tutte le etichette e di sostituire ai richiami di queste,

quindi in tutte le istruzioni GTO, SBR, Dsz, gli indirizzi appena calcolati.

Molto spesso il lavoro di programmazione, specialmente su piccoli dispositivi che non offrono capacità di editing troppo sofisticate, può essere notevolmente alleviato da accorgimenti forse banali, come quello presentato, ma che al momento opportuno risultano di una facilità d'uso e di una comodità estreme.

Sinclair ZX81



a casa vostra subito!

Se volete riceverlo velocemente compilate e spedite in busta il "Coupon Sinclair" e riceverete in OMAGGIO il famoso libro "Guida al Sinclair ZX81" di ben 264 pagine, del valore di L. 16.500.

EXELCO

Via G. Verdi, 23/25
20095 - CUSANO MILANINO (MILANO)

Descrizione	Qt.	Prezzo unitario	Totale L.
Personal Computer ZX81, completo di manuale originale Inglese e cavetti di collegamento al televisore e registratore.		145.000	
Personal Computer ZX81, con alimentatore 0,7 A, completo di manuale originale Inglese e cavetti di collegamento al televisore e registratore.		165.000	
Alimentatore 0,7 A - 9 Vc.c.		25.000	
Modulo di espansione di memoria 16K RAM		131.000	
Valigetta con ZX81, stampante, espansione 16K RAM		460.000	
Valigetta con ZX81, stampante, espansione 32K RAM		530.000	
Valigetta con ZX81, stampante, espansione 64K RAM		620.000	
Stampante Sinclair ZX, con alimentatore da 1,2 A		195.000	
Guida al Sinclair ZX81		16.500	

Desidero ricevere il materiale indicato nella tabella, a mezzo pacco raccomandato, contro assegno, al seguente indirizzo:

Nome

Cognome

Via

Città

Data C.A.P.

Partita I.V.A. o, per i privati
Codice Fiscale

Acconto L.

I prezzi vanno maggiorati dell'IVA 18% e di L. 8.000 per il recapito a domicilio

ATTENZIONE!

Tutti i nostri prodotti hanno la garanzia italiana di un anno, data dalla SINCLAIR.

Operazioni, precisioni, approssimazioni

di P. Capobussi

Discorrere di temi quali l'approssimazione numerica riguardante operazioni matematiche, non è certamente quanto di più semplice possa effettuarsi in ambito divulgativo.

Ritengo che le parole spese in questo campo siano in grado di occupare diversi temi composti da centinaia di pagine, come d'altronde è facile verificare con una rapida visita ad una biblioteca di facoltà matematiche. Se semplicemente desideriamo comprendere meglio come opera il nostro quanto mai utile calcolatore tascabile, non necessariamente dobbiamo però inoltrarci in sofisticati concetti di analisi numerica.

Cerchiamo quindi di dare solo una idea di quali sono i fondamenti e del perché, ad esempio, la precisione del nostro calcolatore non può essere assoluta.

In altre parole: fino a che punto è accettabile il risultato "di poco" diverso da zero che alcune calcolatrici, anche sofisticate, visualizzano quando si effettua l'operazione $\cos 45^\circ - \sin 45^\circ$?

I problemi nascono nel momento in cui desideriamo studiare un fenomeno di qualunque natura e in qualsiasi modo esso si presenti. Può essere uno studio di comportamento sociale, di un esperimento chimico, del funzionamento di un impianto di depurazione delle acque o semplicemente della gestione del nostro conto corrente.

Per tale studio è comodo costruire un modello capace di rendere agevole l'analisi del problema esaminato. Tale modello non è altro che una riproduzione di ciò che effettivamente stiamo studiando, eseguita materialmente rimpicciolendo la realtà, o aiutandosi con esperimenti di laboratorio o ancora descrivendo il tutto con differenti operazioni matematiche.

Ecco allora che si affaccia immediatamente il concetto di approssimazione: un modello non è mai la realtà. Necessariamente costituisce una approssimazione del fenomeno che desideriamo approfondire.

Le discordanze rispetto alla realtà, introdotte dalla costruzione del modello stesso, possono evidentemente essere più o meno grandi e significative. A volte saranno trascurabili; altre volte le soluzioni al problema preso in considerazione saranno affette da un errore che per la sua stessa natura risulta essere ineliminabile. Sarà evidentemente compito di chi analizza il problema scegliere un modello opportuno, il più opportuno.

Giacché con il calcolatore tascabile operiamo in campo prevalentemente matematico, cioè trattiamo dei dati legati da relazioni matematiche, il nostro studio dell'iniziale problema diverrà la ricerca del miglior algoritmo che possa dare soluzione al problema stesso.

L'operazione, benché a prima vista

possa sembrare perfetta in ogni aspetto ("la matematica è una scienza esatta"), non evita l'introduzione di due tipi di errore. Il primo è quello dovuto al fatto che la stima numerica del fenomeno esaminato può essere affetta da approssimazioni (come esempio pensiamo a delle pesate con bilance aventi una loro limitata accuratezza, all'uso di tabelle a loro volta affette da errori o simili). Il secondo tipo di errore è quello che riguarda più da vicino il calcolatore, "pocket" o non che sia: è l'errore di troncamento e arrotondamento. Il calcolatore opera, per sua costituzione fisica, con un numero ben limitato di cifre ed inoltre utilizza algoritmi di calcolo che sono spesso necessariamente incompleti (il calcolo delle funzioni trigonometriche, che viene effettuato tramite sviluppi in serie, è un esempio tipico inerente le calcolatrici scientifiche: di per se stesso infatti il numero degli elementi della serie dovrebbe essere infinito....).

Da un punto di vista pratico, possiamo semplificare la visione del problema affidandoci ad un piccolo esempio.

Fino a che eseguiamo una operazione matematica a mano, e c'è da sperare che avvenga ancora almeno qualche volta, possiamo richiedere una precisione definita ed ottenere così risultati corretti alla stabilita cifra.

Consideriamo l'operazione

1.000.000.000 + 1. Possiamo tranquillamente scrivere su un pezzo di carta:

$$\begin{array}{r} 1.000.000.000 + \\ 1 = \end{array}$$

$$1.000.000.001$$

ottenendo così un risultato esatto. Stiamo operando in quella che viene definita rappresentazione in virgola fissa, ovvero eseguiamo l'operazione incolonnando le cifre e senza spostare la virgola. Negli elaboratori però, senza entrare nel dettaglio dell'operatività reale, i numeri vengono rappresentati in virgola mobile, forma che comprende una parte chiamata mantissa ed una parte chiamata esponente.

Il numero è quindi rappresentato come $M \times 10^E$ ove M è appunto la mantissa ed E l'esponente di 10 che, nella pratica, dice dove è posizionata la virgola. Supponiamo che il nostro calcolatore possa trattare 1 cifra di esponente e 9 di mantissa. I due numeri, nella rappresentazione del calcolatore, si potrebbero allora visualizzare nel modo che il seguente schema esprime figurativamente:

$$\begin{array}{rcl} 10^9 & = & 1.000000000 \quad 9 \\ 1 & = & 0.0000000001 \quad 9 \end{array}$$

$$10^9 = 9 \quad 1.000000000$$

ottenendo così il risultato $10^9 + 1 = 10^9$, il quale non deve scandalizzare, giacché il numero 1 è stato perso per avere superato le capacità fisiche della macchina.

Quanto detto in modo molto semplificato, trova riscontro nei "trucchi" che vengono adottati dai costruttori per limitare, nell'uso, gli inconvenienti di queste inevitabili imprecisioni.

Spesso, ad esempio, il numero di cifre visualizzate è in realtà solo una parte delle cifre per le quali la macchina è in grado di lavorare. Così nascono le cosiddette "cifre di scorta". Non sono altro che cifre non visibili all'operatore che sono ugualmente considerate dalla calcolatrice. Ciò fa sì che l'errore si propaghi solo a queste cifre non visibili (spesso tre), non disturbando, se non in casi eccezionali, il contenuto del display,

che è poi il numero considerato dallo utilizzatore.

Di norma questa diffusa caratteristica degli attuali calcolatori tascabili è da ricordare quando si debbano realizzare dei confronti tra valori numerici derivanti da operazioni differenti. All'apparenza identici, tali valori possono invece "nascondere" nelle proprie cifre di scorta delle differenze anche piccole, ma determinanti ai fini del risultato del confronto. È il caso dell'esempio iniziale citato che richiedeva la sottrazione di $\cos 45^\circ - \sin 45^\circ$. Sarà quindi necessario operare gli opportuni "tagli" ai numeri considerati, tramite differenti tecniche che prevedano la determinazione del numero delle cifre da prendere in esame con cancellazione completa di quelle che vanno eliminate. A volte, nei calcolatori programmabili, semplici sequenze di istruzioni automaticamente azzerano solamente le citate "cifre di scorta".

Comunque quando, alla sequenza di tasti 99999999 Y^x 1 la risposta è 99999998 o peggio, è bene che vi aggiorniate al più presto. Gli attuali microprogrammi che effettuano tali operazioni sono molto più efficienti e qualunque calcolatrice scientifica di recente realizzazione è in grado di dare risposte più accurate. D'altronde è anche per questi motivi che si possono trovare calcolatrici sulle bancarelle dei mercati a prezzi non superiori al costo di una rivista specializzata, quando, a parità di funzioni, marche più prestigiose sono acquistabili, per ora, solo spendendo almeno il doppio.

È comunque indubbio, se ancora qualcuno non l'avesse arguito, che l'uso di calcolatori tascabili, qualsiasi sia l'astuto linguaggio adottato, permette di utilizzare le potenti caratteristiche tipiche delle calcolatrici, quindi di macchine "dedicate ai conti", ovvero capaci di trattare cifre estreme come possono essere un 3500×10^{-99} (!!) o un 75×10^{99} , con un "range" dall'infinitamente grande all'infinitamente piccolo, largamente superiore alle pur notevoli possibilità che offrono molti dei personal odierni.



COMPUTER CLUB TI 99



200 programmi disponibili gratuitamente

- convenzioni agevolate per l'acquisto del tuo home computer
- aiuto all'utilizzo dell'home computer e tanti altri vantaggi che scoprirai associandoti

RIVENDITORI CONVENZIONATI

COMPUTERWORLD - Tel. 06/460818
Via del Traforo, 137 - 00100 ROMA

ESSEMMECI - Tel. 0746/44704
Via delle Orchidee, 19 - 02100 RIETI

COMPUTATA - Tel. 02/545560
Via Botta, 16 - 20135 MILANO

MED - Tel. 0737/3329
Via Venanzi, 11-13 - 62032 CAMERINO (MC)

A TRE - Tel. 0424/25105
Piazzale Firenze, 23

36061 BASSANO DEL GRAPPA (VI)

TECNIPOVAS COMPUTER srl - EDP SHOP
Via Emilia, 36 - 56100 PISA

Tel. 050/502516

COMPUTER CENTER - Tel. 010/300797
Corso Gastaldi, 77/R - 16131 GENOVA

CENTRO DIFFUSIONE MICRO COMPUTER
Via Trento, 42B - 27023 VICEVANO (PV)

MEV system - Tel. 0461/24886
Via Grazioli, 59 - 38100 TRENTO

LEUCI SISTEMI - Tel. 080/902582
Via A. Fighera, 53

74015 MARTINA FRANCA (TA)

VISICOM computer - Tel. 0961/41673
Via Menniti Ippolito, 10 - 88100 CATANZARO

FRANCO - GIOCHI INTELLIGENTI
Corso Fogazzaro, 174

36100 VICENZA - Tel. 0444/42678

SECA - Tel. 0883/44508
Via Postumia, 21 - 70059 TRANI (BA)

G.E.M.E. - Tel. 0963/44655
Via della Pace, 11 Trav. 6

88018 VIBO VALENTIA (CZ)

COMPUTER SHOP - Tel. 095/441620
Via V. E. Orlando, 164-166 - 95127 CATANIA

IMPEL - Tel. 0522/43745
Viale Isonzo, 11A - 42100 REGGIO EMILIA

IMPEL - Tel. 059/225819
Viale Emilia est, 16 - 41100 MODENA

F.I.I. BRENNIA snc - Tel. 031/540096
Via Giordano Bruno, 3 - 22100 COMO

MASH COMPUTER SYSTEM - Tel. 0382/37300
Via Strada Nuova, 86 - 27100 PAVIA

Entra anche tu a far parte della famiglia internazionale degli utenti di Home Computer TI

Computer Club TI 99
Via delle Orchidee n. 19
Tel. 0746/44704-5
02100 RIETI

☐ Sono interessato a

☐ «Computer Club TI 99»

☐ TI-99/4A

Nome e cognome _____

Via _____ cap. _____

Città _____

Telefono _____

Ritagliare e spedire a
«Computer Club TI 99»
Via delle Orchidee n. 19
Tel.: 0746/44705

Istogramma universale per PC-1211/TRS80 pocket

a cura della Redazione

Un'utile e simpatica utilizzazione dei pocket in Basic è certamente quella che permette di tenere sotto controllo dei propri dati gestionali, ad esempio le vendite di un prodotto. Questo programma traccia un istogramma, ovvero un grafico a canne d'organo, sulle ascisse del quale compaiono i mesi dell'anno, mentre sulle ordinate le percentuali, rispetto al totale, di ciascun valore introdotto.

Si sono volute adottare le percentuali proprio per rendere il programma universale. D'altronde non si poteva pretendere di ottenere un grafico a torta, tipicamente utilizzato per la rappresentazione di dati percentuali, date le notoriamente ridotte capacità della stampatina collegabile al TRS 80 Pocket. Il programma possiede una routine di inizializzazione delle variabili da AS (1) a A\$ (12), poi richieste dalla grafica. Alla linea 70 vengono richiesti i valori di ciascun prodotto, valori che vengono sommati e accumulati nella variabile P. La linea 80 effettua il calcolo della percentuale, arrotondato alla unità più prossima, richiamando ciascun valore e dividendolo per il totale contenuto in P. Da A(21) a A(32) vengono memorizzati i valori così calcolati e alla linea 90 i valori così ottenuti vengono verificati per ricavarne il maggiore, quello da cui inizierà la stampa

```
1: CLEAR : PAUSE
  "ISTOGRAMMI"
5: INPUT "TITOL
  O": S$
10: PRINT " ISTO
  GRAMMA"
20: PRINT "      "
  S$
30: PRINT " "
40: PRINT " %!"
50: FOR M=1 TO 12
  : A$(M) = " "
  NEXT M
60: INPUT "N. DE
  I PERIODI " :
  N
70: FOR M=1 TO N:
  INPUT O: A(M+
  20) = O: P = P + O:
  NEXT M
80: FOR M=1 TO 12
  : A(M+20) = INT
  ((A(M+20)*10
  0/P)+.5)
90: IF A(M+20) > R
  LET R = A(M+20
  )
100: NEXT M
110: FOR Q=R TO 1
  STEP -1
120: FOR M=1 TO 12
130: IF A(M+20) >=
  Q LET A$(M) = "
  *"
140: NEXT M
150: IF Q < 10 THEN
  170
160: PRINT Q; " !";
  A$; B$; C$; D$;
  E$; F$; G$; H$;
  I$; J$; K$; L$:
  GOTO 180
170: PRINT " " : Q;
  " !"; A$; B$; C$;
  D$; E$; F$; G$;
  H$; I$; J$; K$;
  L$:
180: NEXT Q
190: PRINT "-----
  -----"
200: PRINT " M !G
  FMAMGLASOND"
210: BEEP 3: END
```

dell'asterisco più alto dell'istogramma. Tramite i cicli FOR...NEXT modificati alle linee 110 e 120 viene ripetuto il procedimento per tutti i valori, stampando il contenuto delle variabili opportunamente caricate con asterischi, alle linee 160 e 170.

Al termine tre squillanti BEEP indicheranno che l'elaborazione è felicemente terminata.

Naturalmente sono possibili molti miglioramenti, ma già in sé risulta essere un programmino utile in più occasioni!

Calcoli ingegneristici con TI 58/59

Il filtro passabasso Butterworth.

di A. Scifoni

Come è noto un filtro passabasso Butterworth di ordine n è un quadripolo a scala costituito da n fra induttanze longitudinali e capacità trasversali, chiuso a sinistra e a destra da due resistenze supposte eguali γR . Quindi a seconda che n sia dispari o pari abbiamo i due tipi in figura 1 e in figura 2.

Questo filtro è usato quando si desidera avere nella banda trasparente un andamento della attenuazione in più possibile piatto.

Chi ha la calcolatrice programmabile TI58 o TI59 trova disponibile, nel modulo SSS Electrical Engineering, un programma che esegue tale calcolo. Però in esso, per il calcolo dell'ordine n del filtro, si ricorre ad un procedimento grafico (nomogramma di Kawakami) che può essere evitato rendendo tutto il calcolo completamente automatico.

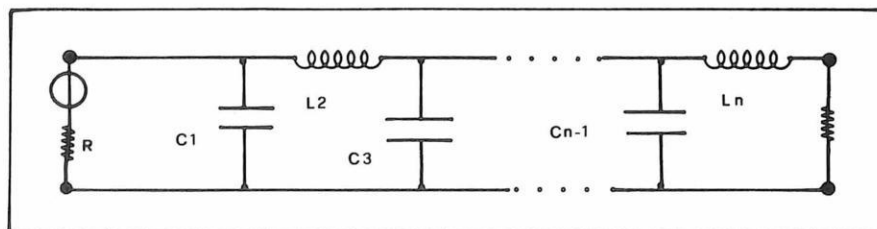


Figura 1. Schema di un filtro Butterworth per n dispari.

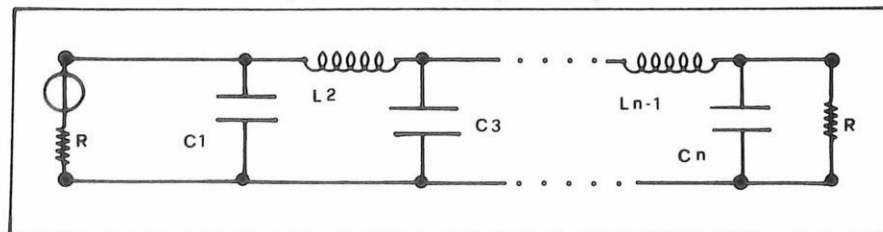


Figura 2. Schema di filtro Butterworth per n pari.

Infatti se la maschera del filtro è quella indicata in figura 3, cioè se l'attenuazione massima che è richiesta nella banda trasparente è

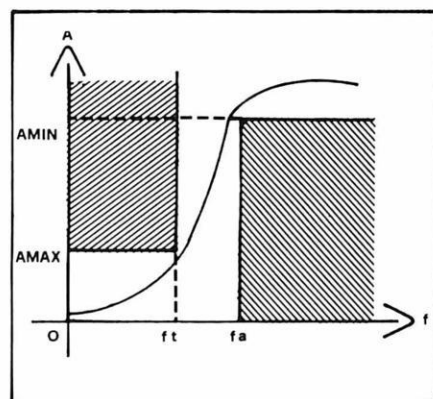


Figura 3. La cosiddetta "maschera" del filtro Butterworth.

$$n_0 = \frac{\log \frac{10^{A_{\min}/10} - 1}{10^{A_{\max}/10} - 1}}{2 \log (f_a/f_t)}$$

Figura 4. Il calcolatore dell'ordine n del filtro viene eseguito tramite questa formula.

$$C_l = \frac{1}{\pi f_t R} \operatorname{sen} \left(\frac{2l-1}{u} 90^\circ \right)$$

$$L_{l+1} = \frac{R}{\pi f_t} \operatorname{sen} \left(\frac{2l+1}{u} 90^\circ \right)$$

$$(l = 1, 2, 3, \dots, u-1)$$

Figura 5. Noto n , applicando questa formula si calcolano i valori delle capacità e delle induttanze.


```

000 76 LBL
001 11 A
002 22 INV
003 57 ENG
004 91 R/S
005 42 STD
006 01 01
007 91 R/S
008 42 STD
009 02 02
010 91 R/S
011 42 STD
012 03 03
013 91 R/S
014 42 STD
015 04 04
016 91 R/S
017 42 STD
018 05 05
019 53 (
020 53 (
021 53 (
022 53 (
023 43 RCL
024 05 05
025 55 ÷
026 01 1
027 00 0
028 54 )
029 22 INV
030 28 LOG
031 75 -
032 01 1
033 54 ) 054 43 RCL
034 55 ÷ 055 04 04
035 53 ( 056 55 ÷
036 53 ( 057 43 RCL
037 43 RCL 058 02 02
038 03 03 059 54 )
039 55 ÷ 060 28 LOG
040 01 1 061 54 )
041 00 0 062 91 R/S
042 54 ) 063 59 INT
043 22 INV 064 42 STD
044 28 LOG 065 06 06
045 75 - 066 69 DP
046 01 1 067 26 26
047 54 ) 068 43 RCL
048 54 ) 069 06 06
049 28 LOG 070 91 R/S
050 55 ÷ 071 00 0
051 02 2 072 42 STD
052 55 ÷ 073 07 07
053 53 ( 074 76 LBL

```

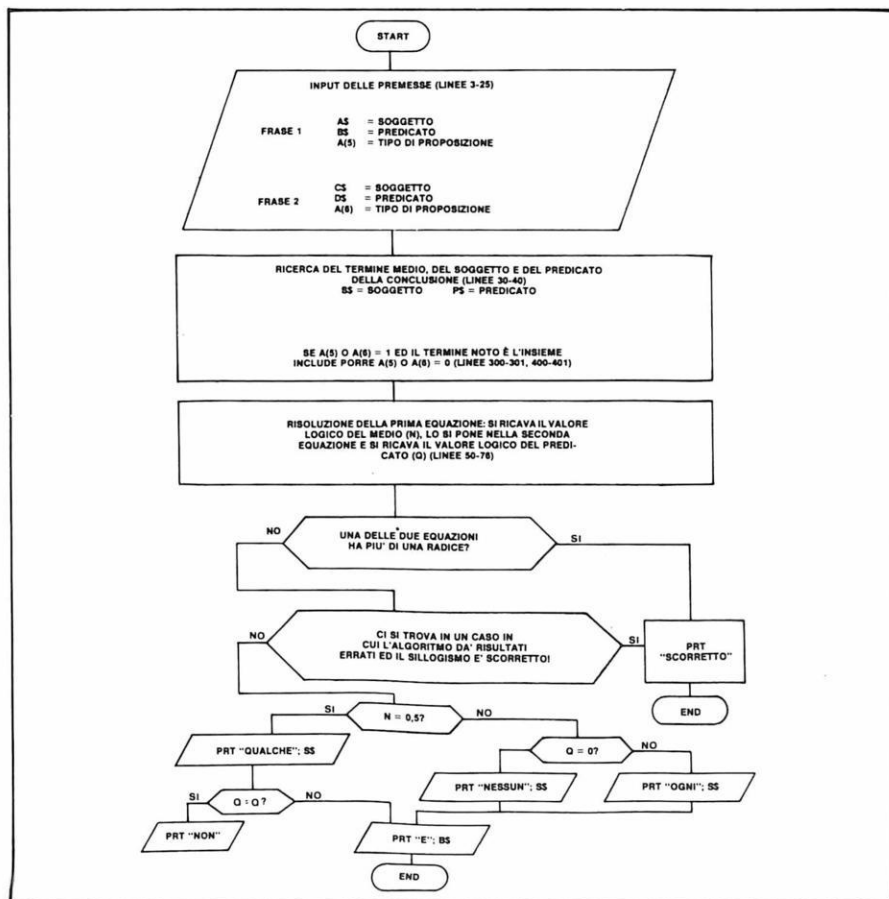


Figura 6. Il diagramma di flusso del programma di calcolo del filtro Butterworth.

```

054 43 RCL 075 12 B 096 43 RCL 117 54 )
055 04 04 076 43 RCL 097 07 07 118 57 ENG
056 55 ÷ 077 01 01 098 75 - 119 91 R/S
057 43 RCL 078 35 1/X 099 01 1 120 12 B
058 02 02 079 42 STD 100 54 )
059 54 ) 080 01 01 101 65 ×
060 28 LOG 081 69 DP 102 09 9
061 54 ) 082 27 27 103 00 0
062 91 R/S 083 43 RCL 104 55 ÷
063 59 INT 084 07 07 105 43 RCL
064 42 STD 085 32 X/T 106 06 06
065 06 06 086 43 RCL 107 54 )
066 69 DP 087 06 06 108 38 SIN
067 26 26 088 22 INV 109 55 ÷
068 43 RCL 089 77 GE 110 89 11
069 06 06 090 11 A 111 65 ×
070 91 R/S 091 53 ( 112 43 RCL
071 00 0 092 53 ( 113 01 01
072 42 STD 093 53 ( 114 55 ÷
073 07 07 094 02 2 115 43 RCL
074 76 LBL 095 65 × 116 02 02

```

ETICHETTE USATE :

```

001 11 A
075 12 B

```

REGISTRI USATI :

```

R1
R2
R3
R4
R5
R6
R7

```

Figura 7. Listato per TI 58/59 adatto a chi non possiede la stampante.

000	76	LBL	053	03	3	106	99	PRT	159	28	LOG	212	00	0
001	11	A	054	05	5	107	02	2	160	75	-	213	00	0
002	98	ADV	055	00	0	108	01	1	161	01	1	214	00	0
003	91	R/S	056	00	0	109	01	1	162	54	)	215	01	1
004	76	LBL	057	07	7	110	03	3	163	55	÷	216	42	STD
005	15	E	058	01	1	111	00	0	164	53	(	217	00	00
006	01	1	059	00	0	112	00	0	165	53	(	218	02	2
007	69	DP	060	00	0	113	07	7	166	43	RCL	219	07	7
008	17	17	061	00	0	114	01	1	167	03	03	220	00	0
009	22	INV	062	00	0	115	69	DP	168	55	÷	221	00	0
010	57	ENG	063	69	DP	116	04	04	169	01	1	222	00	0
011	69	DP	064	04	04	117	69	DP	170	00	0	223	01	1
012	00	00	065	69	DP	118	05	05	171	54	)	224	42	STD
013	01	1	066	05	05	119	91	R/S	172	22	INV	225	09	09
014	04	4	067	91	R/S	120	42	STD	173	28	LOG	226	00	0
015	04	4	068	42	STD	121	04	04	174	75	-	227	42	STD
016	01	1	069	01	01	122	99	PRT	175	01	1	228	07	07
017	03	3	070	99	PRT	123	01	1	176	54	)	229	76	LBL
018	07	7	071	02	2	124	03	3	177	54	)	230	12	B
019	03	3	072	01	1	125	03	3	178	28	LOG	231	43	RCL
020	07	7	073	03	3	126	00	0	179	55	÷	232	01	01
021	01	1	074	07	7	127	02	2	180	02	2	233	35	1/X
022	07	7	075	00	0	128	04	4	181	55	÷	234	42	STD
023	69	DP	076	00	0	129	03	3	182	53	(	235	01	01
024	01	01	077	07	7	130	01	1	183	43	RCL	236	43	RCL
025	03	3	078	01	1	131	07	7	184	04	04	237	00	00
026	05	5	079	00	0	132	01	1	185	55	÷	238	48	EXC
027	04	4	080	00	0	133	69	DP	186	43	RCL	239	09	09
028	03	3	081	69	DP	134	04	04	187	02	02	240	48	EXC
029	03	3	082	04	04	135	69	DP	188	54	)	241	00	00
030	02	2	083	69	DP	136	05	05	189	28	LOG	242	69	DP
031	03	3	084	05	05	137	91	R/S	190	54	)	243	27	27
032	05	5	085	91	R/S	138	42	STD	191	69	DP	244	43	RCL
033	03	3	086	42	STD	139	05	05	192	06	06	245	07	07
034	07	7	087	02	02	140	99	PRT	193	59	INT	246	32	XIT
035	69	DP	088	99	PRT	141	98	ADV	194	42	STD	247	43	RCL
036	02	02	089	01	1	142	03	3	195	06	06	248	06	06
037	02	2	090	03	3	143	01	1	196	69	DP	249	22	INV
038	03	3	091	03	3	144	00	0	197	26	26	250	77	GE
039	00	0	092	00	0	145	01	1	198	03	3	251	11	A
040	00	0	093	01	1	146	69	DP	199	01	1	252	53	(
041	03	3	094	03	3	147	04	04	200	00	0	253	43	RCL
042	03	3	095	04	4	148	53	(	201	00	0	254	07	07
043	04	4	096	04	4	149	53	(	202	69	DP	255	55	÷
044	00	0	097	07	7	150	53	(	203	04	04	256	01	1
045	01	1	098	01	1	151	53	(	204	43	RCL	257	00	0
046	04	4	099	69	DP	152	43	RCL	205	06	06	258	54	)
047	69	DP	100	04	04	153	05	05	206	69	DP	259	59	INT
048	03	03	101	69	DP	154	55	÷	207	06	06	260	42	STD
049	69	DP	102	05	05	155	01	1	208	69	DP	261	08	08
050	05	05	103	91	R/S	156	00	0	209	00	00	262	53	(
051	69	DP	104	42	STD	157	54	)	210	01	1	263	01	1
052	00	00	105	03	03	158	22	INV	211	05	5	264	00	0

Figura 8. Listato per TI 58/59 con stampante PC-100.

```

265 00 0 318 54 )
266 65 x 319 59 INT
267 43 RCL 320 85 +
268 08 08 321 43 RCL
269 85 + 322 09 09
270 01 1 323 85 +
271 00 0 324 43 RCL
272 65 x 325 07 07
273 53 ( 326 54 )
274 01 1 327 69 DP
275 00 0 328 04 04
276 75 - 329 53 (
277 43 RCL 330 53 (
278 08 08 331 53 (
279 54 ) 332 02 2
280 54 ) 333 65 x
281 65 x 334 43 RCL
282 53 ( 335 07 07
283 01 1 336 75 -
284 75 - 337 01 1
285 53 ( 338 54 )
286 53 ( 339 65 x
287 01 1 340 09 9
288 00 0 341 00 0
289 75 - 342 55 ÷
290 43 RCL 343 43 RCL
291 08 08 344 06 06
292 54 ) 345 54 )
293 55 ÷ 346 38 SIN
294 01 1 347 55 ÷
295 00 0 348 89 π
296 54 ) 349 65 x
297 59 INT 350 43 RCL
298 54 ) 351 01 01
299 85 + 352 55 ÷
300 02 2 353 43 RCL
301 65 x 354 02 02
302 53 ( 355 54 )
303 53 ( 356 57 ENG
304 01 1 357 69 DP
305 85 + 358 06 06
306 43 RCL 359 12 B
307 07 07
308 75 -
309 01 1
310 00 0
311 65 x
312 43 RCL
313 08 08
314 54 )
315 55 ÷ 001 11 A
316 01 1 005 15 E
317 00 0 230 12 B

```

ETICHETTE USATE : REGISTRI USATI :

R0	R5
R1	R6
R2	R7
R3	R8
R4	R9

Procedura relativa al programma con stampante

Dopo aver introdotto il programma nella memoria della calcolatrice a partire da un certo passo, normalmente dal passo 000, si batte E (*).

A questo punto vengono stampate l'intestazione BUTTERWORTH P.B. e le richieste dei dati d'entrata sotto forma delle scritte R?, AMAX?, FA?, AMIN?, dopo ognuna delle quali si deve battere il dato d'entrata specificato e successivamente R/S.

Quindi dopo la stampa R? si batte la resistenza R in ohm e R/S, dopo la stampa FT? si batte la frequenza ft in hertz e R/S, dopo la stampa AMAX? si batte l'attenuazione Amax in decibel e R/S, dopo la stampa FA? si batte la frequenza fa in hertz e R/S e infine dopo la stampa AMIN? si batte l'attenuazione Amin in decibel e R/S.

La calcolatrice dopo l'elaborazione stampa i valori d'uscita n_o , n, C1, L2, C3, L4, ... (in farad e in henry) stampando rispettivamente sulle stesse righe a destra le indicazioni NO, N, C1, L2, C3, L4, ... (**).

Dopo la stampa dell'ultimo dato d'uscita, che sarà Cn se n è dispari oppure Ln se n è pari il calcolatore si ferma.

Per predisporlo automaticamente a ricevere una nuova serie di dati d'entrata con le modalità viste si deve battere nuovamente E (*).

All'inizio e alla fine e fra i dati d'entrata e i dati d'uscita sono previste delle interlinee di separazione.

Ad esempio se vogliamo progettare un filtro Butterworth passabasso relativo ai seguenti dati d'entrata:

```

R      = 2000 ohm
ft     = 10000 hertz
Amax   = 3 decibel
fa     = 15000 hertz
Amin   = 30 decibel

```

alla fine dell'elaborazione avremo lo stampato di figura 10, dal quale si vede immediatamente che con le specifiche di progetto indicate abbiamo un filtro costituito da 9 componenti e precisamente da:

```

C1 = C9 = 2,7636966 nanofarad
L2 = L8 = 31,831989 millihenry
C3 = C7 = 12,191976 nanofarad
L4 = L6 = 59,82269 millihenry
C5 = 15,915494 nanofarad

```

(*) Oppure si batte A e dopo un'interlinea si batte R/S, oppure se si è registrato il programma a partire dal passo 000 si batte RST e R/S e dopo un'interlinea si batte R/S.

(**) Le scritte C1, L2, L3, L4, ... sono esatte fino all'eventuale valore C89, largamente sufficiente per gli scopi pratici, essendo corrispondente ad un filtro di ben 89 componenti. La stampa di ogni riga richiede circa 8 secondi.

"Amax" e l'attenuazione minima che è richiesta nella banda opaca è "Amin", essendo "ft" la frequenza di taglio e "fa" la frequenza estrema della banda di transizione, l'ordine "n" del filtro viene determinato calcolando prima " n_o " con la formula di figura 4 e poi prendendo per n l'intero più vicino ad n_o maggiore di esso, cioè $n = \text{INT } n_o + 1$.

Noto allora n si calcolano le capa-


```

BUTTERWORTH P.B
R ?
2000. FT ?
10000. AMAX?
3. FA ?
15000. AMIN?
30.
8.52293278 NO
9. N
2.7636966-09 C 1
31.830989-03 L 2
12.191976-09 C 3
59.82269-03 L 4
15.915494-09 C 5
59.82269-03 L 6
12.191976-09 C 7
31.830989-03 L 8
2.7636966-09 C 9

```

Figura 10. Tramite il programma che gira con la stampante viene prodotta l'uscita mostrata.

cità e le induttanze del filtro mediante le formule di figura 5.

In fig. 6 è invece illustrata il flow-chart del programma di calcolo di questo filtro secondo quanto richiamato, e può essere utile per codificarlo anche in altri linguaggi, oltre che in S.O.A.

In fig. 7 è illustrata la relativa codificazione in S.O.A. in versione senza stampante per la TI58 e la TI59 e in fig. 8 infine è illustrata la relativa codifica per la TI58 o la TI59 in cui è previsto l'uso della stampante.

Mentre il primo programma è costituito da 121 passi ed è quindi possibile registrarlo in memoria volta per volta, per il secondo sono richiesti 360 passi e quindi per introdurlo in memoria occorre un certo tempo e attenzione. È comodo usarlo con la TI59 registrandolo una volta per tutte su scheda magnetica.

Comunque, data la capacità di memoria richiesta, nulla vieta di memorizzarlo con un po' di pazienza anche sulla TI58C.

Sinclair Spectrum



a casa vostra subito!

Se volete riceverlo velocemente compilate e spedite in busta il "Coupon Sinclair" e riceverete in OMAGGIO il famoso libro "Guida al Sinclair ZX Spectrum" di ben 120 pagine, del valore di L. 22.000.

EXELCO

Via G. Verdi, 23/25
20095 - CUSANO MILANINO (MILANO)

Descrizione	Qt.	Prezzo unitario	Totale L.
Personal Computer ZX Spectrum 16K RAM con alimentatore, completo di manuale originale Inglese e cavetti di collegamento.		360.000	
Personal Computer ZX Spectrum 48K RAM con alimentatore, completo di manuale originale Inglese e cavetti di collegamento.		495.000	
Kit di espansione 32K RAM.		Annunciato	
Stampante Sinclair ZX, con alimentatore da 1,2 A.		195.000	
Guida al Sinclair ZX Spectrum.		22.000	
Cassetta programmi dimostrativi per il rapido apprendimento alla programmazione e utilizzo dello ZX Spectrum.		48.000	

Desidero ricevere il materiale indicato nella tabella, a mezzo pacco raccomandato, contro assegno, al seguente indirizzo:

Nome
 Cognome
 Via
 Città
 Data C.A.P.
 Partita I.V.A. o, per i privati Codice Fiscale
 Acconto L.

I prezzi vanno maggiorati dell'IVA 18% e di L. 8.000 per il recapito a domicilio

ATTENZIONE!

Tutti i nostri prodotti hanno la garanzia italiana di un anno, data dalla SINCLAIR.

Procedura relativa al programma senza stampante

Dopo aver introdotto il programma nella memoria della calcolatrice a partire da un certo passo, normalmente dal passo 000, si batte A (*) e successivamente si battono:

la resistenza R in ohm e R/S,
la frequenza f_t in hertz e R/S,
l'attenuazione A_{max} in decibel e R/S,
la frequenza f_a in hertz e R/S,
l'attenuazione A_{min} in decibel e R/S.

la calcolatrice, dopo l'elaborazione, visualizza sul display prima n_o , poi battendo R/S visualizza n , poi battendo dopo ogni numero R/S visualizza nell'ordine, rispettivamente in farad ed in henry, C1, L2, C3, L4, ... fino all'ultimo componente che è, per n dispari, una capacità (Cn) e per n pari una induttanza (Ln) (**).

Dopo aver battuto l'ultimo R/S il calcolatore si predispone automaticamente a ricevere una nuova serie di dati d'entrata R , f_t , A_{max} , f_a , A_{min} secondo le modalità viste.

Da notare che si è preferito usare il metodo di battere dopo ogni dato R/S invece di usare delle etichette A, B, C, ... per una maggiore velocità di digitazione anche se ogni volta occorre ripetere l'intera serie di dati d'entrata.

Da notare inoltre che le uscite sono date in notazione tecnica, cioè espresse con potenze di dieci a esponente multiplo di tre, per cui è immediato passare da farad a millifarad, nanofarad e microfarad e da henry a millihenry e microhenry.

Questa notazione è opportuna specialmente per le capacità, che essendo normalmente espresse da numeri molto piccoli se misurate in farad, potrebbero essere date con un piccolo numero di cifre significative (ad esempio se $C1 = 2,764$ nanofarad avremmo il numero 0,0000000028 con due sole cifre significative).

Da notare infine che in uscita è dato anche il numero n_o oltre che n . Questo perché in alcuni casi, e precisamente quando n_o è di poco superiore ad un numero intero, modificando leggermente le specifiche di progetto, cioè i valori numerici dei dati d'entrata, si può risparmiare un componente.

Ad esempio se abbiamo come dati d'entrata

$R = 3000$ ohm
 $f_t = 450$ hertz
 $A_{max} = 5$ decibel
 $f_a = 850$ hertz
 $A_{min} = 20$ decibel

otterremo all'uscita i valori

$n_o = 3,006309161$
 $n = 4$
 $C1 = 90,231052$ nanofarad
 $L2 = 1,9605333$ henry
 $C3 = 217,83703$ nanofarad
 $L4 = 812,07947$ millihenry

Essendo il valore di n_o leggermente superiore a 3 si può cercare di modificare leggermente i dati d'entrata per ottenere un n_o minore o eguale a 3.

Supponendo di poter scegliere per A_{min} un valore di mezzo decibel inferiore, cioè prendendo $A_{min} = 19,5$ decibel si ottiene all'uscita:

$n_o = 2,914827573$
 $n = 3$
 $C1 = C3 = 117,89255$ nanofarad
 $L2 = 2,1220659$ henry
e quindi si è risparmiata una induttanza da circa un henry abbastanza costosa.

Naturalmente se si vuole per questo programma la stampa dei dati d'entrata e d'uscita senza ricorrere al secondo programma più sofisticato basta sostituire agli opportuni R/S dei PRT ed eventualmente degli ADV per spaziare opportunamente le serie di dati.

Con queste modifiche gli stampati relativi ai due esempi precedenti sono mostrati in figura 9.

3000.	3000.
450.	450.
5.	5.
850.	850.
20.	19.5
3.006309161	2.914827573
4.	3.
90.231052-09	117.89255-09
1.9605333 00	2.1220659 00
217.83703-09	117.89255-09
812.07947-03	

(*) Oppure se si è registrato il programma a partire dal passo 000 si batte RST e R/S.

(**) La stampa di ogni riga richiede circa 4 secondi.

Figura 9. Sostituendo alla versione senza stampante gli R/S con dei PRT, si ottengono i risultati stampati come in figura.

Brevi ricette per ottimizzare

di Paolo Capobussi

La prima necessità di coloro che si dedicano ai pocket in BASIC è, necessariamente, quella di risparmiare memoria di programma. Generalmente, infatti, l'allocazione della memoria è dinamica, nel senso che se diamo molto spazio al programma diminuisce considerevolmente l'area disponibile ai dati, e viceversa.

Avendo a disposizione molta memoria ciò non costituirebbe un grosso problema, ma su pocket come il Casio FX-702P o similari, che non possono essere espansi a più di circa 2,5 Kbyte, una ottimizzazione dei passi è fortemente consigliata.

Una seconda necessità è, certamente, quella di guadagnare tempo di elaborazione. Per macchine basate su circuiti C-MOS il problema della velocità di elaborazione è spesso un fattore di pesante limitazione all'applicabilità di tali computer tascabili.

Risulta difficile prevedere tutti i casi in cui è possibile una modifica strutturale al programma per renderlo più veloce e più compatto, ma certamente si può centrare il problema puntando lo sguardo su un tipo di macchina e per alcune situazioni tipiche.

Le note su cui vorremmo porre la nostra attenzione riguardano il pocket Casio FX-702P, ma i ragionamenti sono spesso applicabili agli altri piccoli pocket. È, d'altronde, impensabile arrivare ad una generalizzazione dell'analisi, diversamente da quanto si potrebbe fare per macchine più potenti, in quanto per i pocket e le cosiddette calcolatrici programmabili molto spesso un solo

```
LIST #7
5 $="1234567890":
WAIT 0:B=0
10 IF KEY="" THEN
10
20 FOR I=1 TO 10:I
F KEY=MID(I,1):
NEXT I:GOTO 10
30 B=B+1:PRT I:GOT
0 10
```

```
LIST #4
5 A=130:WAIT 0:B=
0
20 IF KEY="1":I=1:
GOTO A
30 IF KEY="2":I=2:
GOTO A
40 IF KEY="3":I=3:
GOTO A
50 IF KEY="4":I=4:
GOTO A
60 IF KEY="5":I=5:
GOTO A
70 IF KEY="6":I=6:
GOTO A
80 IF KEY="7":I=7:
GOTO A
90 IF KEY="8":I=8:
GOTO A
100 IF KEY="9":I=9:
GOTO A
110 IF KEY="0":GOTO
20
120 I=0
130 B=B+1:PRT I:GOT
0 20
```

Fig. 1

```
LIST
50 X=2:FOR I=1 TO
100:Y=X+2:NEXT
I:PRT "FINE"

LIST
50 X=2:FOR I=1 TO
100:Y=X*X:NEXT
I:PRT "FINE"
```

Fig. 2

```
LIST #4
10 FOR I=1 TO 1000
:GSB 100
20 NEXT I:PRT "STO
P"
100 RET
```

```
LIST #7
10 FOR I=1 TO 1000
:GOTO 100
20 NEXT I:PRT "STO
P"
100 GOTO 20
```

Fig. 3

```
LIST #7
10 WAIT 50:VAC :IN
P "A=",A,"B=",B
,"C=",C,"D=",D
20 X=A:GSB 50:A=Y:
X=B:GSB 50:B=Y:
X=C:GSB 50:C=Y:
X=D:GSB 50:D=Y
30 PRT "A=",A,"B=",
:B,"C=",C,"D=",
D:GOTO 10
50 Y=SIN X+2+COS X
/2
60 RET
```

```
LIST #4
10 VAC :WAIT 20:IN
P "A=",A,"B=",B
,"C=",C,"D=",D
20 X=A:$="A":GSB 5
0:X=B:$="B":GSB
50:X=C:$="C":G
SB 50
30 WAIT 50:X=D:$="
D":GSB 50:GOTO
10
50 Y=SIN X+2+COS X
/2
60 PRT $:"=":Y:RET
```

Fig. 4

```
P0: 40 STEPS
10 GOTO 1000
20 GOTO 1000
30 GOTO 1000
40 GOTO 1000
50 GOTO 1000
```

```
P1: 32 STEPS
10 Z=1000:GOTO Z
20 GOTO Z
30 GOTO Z
40 GOTO Z
50 GOTO Z
```

Fig. 5

```
P0: 45 STEPS
10 VAC
20 INP "A",A
30 IF FRAC (A/2)=0
THEN 50
40 A=A+1
50 PRT A:GOTO 10
```

```
P1: 40 STEPS
10 VAC
20 INP "A",A
30 IF FRAC (A/2)*0
:A=A+1
40 PRT A:GOTO 10
```

Fig. 6

```
P0: 50 STEPS
10 WAIT 5:PRT "5":
FOR I=1 TO 60:N
EXT I:PRT "QUES
TA E UNA PRO123
456789"
```

```
P0: 41 STEPS
10 WAIT 5:PRT "5":
FOR I=1 TO 60:N
EXT I:PRT "QUES
TA E UNA PRO"
```

Fig. 7

passo guadagnato significa la soluzione di problemi altrimenti irrisolvibili. Diventa quindi predominante la necessità di sfruttare ogni caratteristica non tanto del linguaggio utilizzato, quanto del particolare sistema operativo supportato dalla macchina stessa, quindi molto diverso in ogni modello.

Innanzitutto è sempre bene puntare l'attenzione al problema effettivo che si vuole risolvere. Difficilmente, infatti, il problema della velocità si sposa con quello della compattezza. Cercare la strada per risparmiare in passi di programmi significa infatti, molto spesso, arrivare a soluzioni che si piegano su se stesse, con loop intrecciati tra loro e conseguente minore linearità di elaborazione, quindi minore velocità. Consideriamo l'esempio di figura 1.

Riporta un programma in due versioni. L'una permette di introdurre dei dati numerici tramite l'uso della funzione KEY, ed utilizza solo 76 passi. L'altra ancora utilizza la

stessa funzione e permette ugualmente l'introduzione di dati numerici, ma utilizza 136 passi.

Il funzionamento della prima è molto semplice. Caricata la stringa \$ con la serie delle cifre decimali, si mette in attesa della pressione di un tasto, continuando a girare sulla linea 10. Come un tasto viene premuto è confrontato con la serie dei numeri decimali precedentemente memorizzata, tramite la funzione MID che estrae cifra per cifra dalla stringa e la confronta con quella introdotta. In caso di uguaglianza la cifra premuta è visualizzata sul display, mentre in caso contrario torna in attesa.

Un contatore realizzato tramite la variabile B permette di contare quante introduzioni valide sono state effettuate.

La seconda routine realizza la stessa funzione, come abbiamo detto, ma tramite l'uso di confronti separati per ogni cifra.

Provando questi programmi si ot-

tiene un guadagno di tempo che raggiunge addirittura il 75%, della seconda routine rispetto la prima. Pur variando in funzione del numero introdotto, è evidente che il tempo minore lo si ottiene con la routine più lunga. Infatti sono richieste alla macchina un numero maggiore di operazioni più semplici. Per curiosità si può dire, ad esempio, che in un minuto la prima routine può accettare un massimo di 116 cifre "4" (tenendo premuto costantemente il tasto), contro le 153 cifre "4" introducibili tramite la seconda routine. Se volete ripetere la prova, tenete presente che tali numeri sono facilmente ricavabili richiamando, dopo un minuto, il contenuto della variabile B.

Pur non riguardando l'ottimizzazione dei tempi o l'occupazione di memoria, di queste routine è interessante notare l'uso della funzione KEY. Non è sempre necessario, infatti, utilizzare una variabile apposita per memorizzare il contenuto prelevato da tastiera tramite KEY. Ovvero non è sempre necessario utilizzare lo statement $K\$ = KEY$ per memorizzare un singolo tasto.

La funzione KEY è, infatti, considerabile come associata ad una variabile a sé stante, il cui contenuto è testabile esattamente come il contenuto di una variabile.

Se utilizziamo la funzione di elevamento a potenza, i tempi di esecuzione si allungano notevolmente rispetto all'uso diretto di moltiplicazioni successive.

Qualche semplice prova con il cronometro alla mano vi convincerà che usando variabili separate, a parità del grado di elevamento, il tempo rimane praticamente costante anche cambiando la cifra da elevare; mentre usando la funzione di elevamento a potenza il tempo varia ed è comunque superiore.

Anche in tal caso il guadagno di passi è maggiore là dove i tempi sono invece sfavorevoli.

Le subroutine sono generalmente da favorire rispetto ai richiami diretti o indiretti eseguiti tramite l'istruzione GOTO.

Provando i programmi di figura 3 è facile cronometrarne i tempi di esecuzione. Per il primo si superano

Figura 1. La prima routine di introduzione dati utilizza una sola volta l'istruzione KEY, memorizzando i termini di confronto nella variabile \$. La seconda, pur occupando un numero maggiore di passi, permette però una introduzione più veloce.

Figura 2. L'elevamento a potenza è eseguibile anche tramite una serie di moltiplicazioni, con risparmio di tempo di esecuzione, ma conseguente aumento del numero di passi occupati.

Figura 3. La prima routine mostra l'uso di un GOTO, sostituito nella seconda con un richiamo di subroutine.

Figura 4. Due modi di visualizzare dei dati, risultati di uno stesso calcolo. Nel primo caso prima avviene il calcolo, poi la visualizzazione di tutti i risultati. Nel secondo caso si sfrutta il tempo di elaborazione per tenere visualizzato il risultato precedente.

Figura 5. Un confronto tra indirizzamento diretto e indiretto. In questo semplice caso si risparmiano 8 passi di programma.

Figura 6. L'uso accorto dei confronti permette, spesso, notevolissimi risparmi di passi. In questa figura si nota subito una particolarità di questi pocket: la parola chiave THEN deve essere sostituita dal punto e virgola se segue un'altra istruzione invece che un numero di linea a cui saltare.

Figura 7. Pur apparentemente identici, questi listati mostrano come si possano introdurre altri caratteri su una linea supposta piena, semplicemente eliminando gli spazi prima e dopo il numero di linea e tra ciascuna parola chiave. In questo caso, puramente dimostrativo, normalmente non saremmo riusciti neppure a completare la parola PROVA, che infatti è rimasta come PRO. Mentre compattando la linea è stato possibile introdurre addirittura altri nove caratteri, come stanno a dimostrare i numeri appositamente aggiunti dopo PRO.

i 70 secondi, mentre il secondo non richiede più di 52 secondi. L'uso dei GOTO richiede sempre la ricerca dello statement corrispondente a partire dall'inizio della memoria di programma, mentre l'uso di RET implica l'utilizzo di un indirizzo che la macchina ha memorizzato già nello stack, con ovvio risparmio di tempo di ricerca.

L'uso di sottoprogrammi permette anche di realizzare strutture come quelle di figura 4. L'utilità della routine è molto semplice: visualizzare dei risultati che derivano dall'esecuzione di uno stesso calcolo ripetuto più volte con dati differenti. Se utilizziamo una struttura come quella rappresentata nel primo caso, i valori richiesti vengono dapprima tutti calcolati e poi visualizzati in ordine.

Quindi esiste un tempo di calcolo e un tempo di visualizzazione regolato dal valore dato all'istruzione WAIT (che è da una istruzione tipica della FX-702P Casio che permette di regolare la durata di visualizzazione dello statement PRINT).

La seconda routine è invece strutturata in modo tale che i risultati del calcolo vengono visualizzati immediatamente dopo il calcolo stesso, ma per una durata che viene regolata dal fatto che durante la visualizzazione il calcolo viene fatto proseguire al successivo. Si sfrutta quindi il tempo di elaborazione dei dati per visualizzare il dato precedente. Naturalmente ciò, essendo molto semplice il calcolo effettuato, porterebbe a tempi di visualizzazione più brevi dei precedenti. Per ovviare a questo si introduce ugualmente un WAIT, ma di valore inferiore. L'uso dei dati come GOTO o GSB è possibile anche nel cosiddetto modo indiretto. In tal caso si utilizza una variabile quale argomento dell'istruzione di salto. Dando differenti valori alla variabile stessa si potrà pilotare il salto in posizioni differenti.

Tale metodo è ottimo quando si vogliano risparmiare passi di programma. Se è vero, infatti, che un numero di linea comunque lungo (da 1 a 9999) occupa sempre 3 soli passi, quando il numero di linea è argomento di una istruzione occupa tanti passi quante sono le cifre di cui è composto. Quindi converrebbe

utilizzare sempre numeri di linea bassi per routine richiamate con frequenza. A ciò si può però ovviare proprio con l'indirizzamento indiretto. Infatti, memorizzato il numero in una variabile, è poi sufficiente utilizzare il nome di tale variabile che occupa un solo passo quale argomento dell'istruzione.

Si veda al proposito il chiaro esempio di figura 5.

Per sfruttare meglio gli indirizzamenti è bene tenere presente le capacità matematiche di questi pocket tascabili. Meglio è scrivere, infatti, GOTO 1E3 piuttosto che GOTO 1000, oppure FOR J = 1 TO 1E5 piuttosto che FOR J = 1 TO 100000. Si ha un evidente risparmio di passi.

Altre abbreviazioni possibili sono concernenti le variabili. Per le prime dieci disponibili è più semplice scrivere A1 piuttosto che non A(1) ove si sprecano due parentesi. Oppure A\$(9) è equivalente a A9\$.

La figura 6 mostra, invece, come si possano risparmiare 5 passi ponendo attenzione ai criteri di confronto, e sfruttando le possibilità dell'istruzione IF, la quale dopo un punto e virgola permette di eseguire qualunque altra istruzione differente da THEN.

La visualizzazione dei dati è comunque il punto generalmente più critico di questi piccoli elaboratori. Possedendo una sola linea di visualizzazione a cristalli liquidi, messaggi più lunghi del massimo numero di caratteri visibili contemporaneamente obbligano la macchina a far scorrere la scrittura a velocità che consentano la corretta lettura, generalmente molto lenta. Ciò comporta dei rallentamenti che molto spesso finiscono con l'innervosire lo stesso utilizzatore. Piuttosto che ricorrere ogni volta alla pressione del tasto CONT, che sulla FX-702P non solo permette di proseguire un programma interrotto con STOP, ma anche di by-passare ogni istruzione di pausa voluta dal programmatore con WAIT o inserita dal sistema operativo nel caso, appunto, della visualizzazione, è bene ricorrere a metodi preventivi. Sarà quindi cura del programmatore evitare i messaggi più lunghi di 20 caratteri

(o di quanti caratteri massimi sono visualizzabili sul display senza scorrimento) quando si usi l'istruzione PRINT. Mentre dovrà sempre tener presente che viene inserito un punto di domanda ogni volta che usa una istruzione INP, quindi limitando di un carattere la lunghezza massima.

Se comunque la necessità è quella di realizzare un messaggio lungo, ma su quella linea sembra impossibile introdurre alcunché, e spesso utile ricorrere ad una scrittura compatta delle linee stesse.

Quando si scrive una nuova linea, infatti, vengono automaticamente inseriti degli spazi tra le parole chiave, le variabili e ogni altro simbolo interpretabile indipendentemente dal sistema operativo. Anche prima del numero di linea viene introdotto uno spazio. In realtà questa operazione non sarebbe strettamente necessaria, anche se migliora notevolmente la leggibilità.

È quindi sempre possibile eliminare tutti gli spazi in eccesso tramite un accorto uso del tasto C e così avere maggiore spazio per introdurre altri messaggi o nuove istruzioni.

Sia comunque chiaro che tale metodo a nulla vale per recuperare dei passi di programma. Gli spazi introdotti dalla macchina stessa, infatti, non sono comunque conteggiati come occupazione di memoria. Inoltre il listato effettuato sulla stampante non permette di rendersi conto del compattamento effettuato, in quanto tutti gli spazi vengono automaticamente ripristinati. Un esempio si ha in figura 7. Nel primo caso un compattamento ha reso possibile aggiungere i numeri da 1 a 9, altrimenti non inseribili come è facile verificare dalla stessa figura nel secondo listato.

Tutti i piccoli trucchi descritti sono certamente utili per il tipico lavoro da certoso che la programmazione degli elaboratori tascabili ha sempre richiesto, ma sempre meno richiederà in futuro all'aumentare della memoria disponibile. Sottolineano comunque un grosso svantaggio sempre in agguato: la difficoltà di documentare un programma realizzato su questi piccoli e infernali marchingegni.

Quadratini e lacune per TI 58-59

Un gioco di abilità e di
pazienza per chi non
possiede la PC-100A

di F. Carbone

Chi non ricorda che alcuni anni fa, prima dell'avvento del celebre cubo magico, era in commercio qualcosa ad esso simile ma in sole due dimensioni? Si trattava di una tavoletta di plastica sulla quale erano incastrati, in modo da potersi muovere l'uno rispetto all'altro, quindici quadratini numerati e ordinati in serie crescente su quattro file, così che nella fila più bassa ne erano presenti solo tre, col quattordicesimo e il quindicesimo scambiati di posto. Il gioco consisteva nel far andare, mediante successivi spostamenti, il "14" al posto del "15". Ma per far ciò era inevitabile che i numeretti si disordinassero del tutto e non era cosa facile ottenere un quadrato completamente ordinato.

Il gioco che qui viene proposto in versione elettronica è una riduzione a soli nove quadrati rispetto all'originale. La calcolatrice è qui solo il supporto materiale del gioco (utilizzo passivo del calcolatore), sostituendosi alla classica tavoletta. Tuttavia l'impossibilità di spostare più quadretti in una sola volta, ed una diversa interpretazione del principio base del gioco, possono renderne

l'esecuzione più interessante e forse più difficoltosa.

Per realizzare il programma si è partiti dal seguente concetto: quando spostiamo un quadretto nella zona vuota, in effetti eseguiamo due azioni contemporanee: mettiamo un oggetto in uno spazio vuoto e togliamo l'oggetto dal posto che occupava (potremmo dire che mettiamo lo spazio vuoto al posto dell'oggetto). Effettuiamo cioè uno scambio di posto tra due oggetti, il quadratino e lo spazio vuoto. Spostare allora un quadratino verso il basso equivarrà a spostare lo spazio vuoto verso l'alto. Per chi di voi ha familiarità con i semiconduttori, il concetto richiama molto bene alla mente lo spostamento di elettroni e lacune.

Nel nostro programma si parte da una configurazione che vede gli otto quadretti completamente ordinati e lo spazio vuoto simboleggiato dal numero 9. I tasti A, B, C e D faranno spostare il nove rispettivamente verso l'alto, il basso, a sinistra e destra. Bisogna quindi disordinare i numeri prima di passare al gioco vero e proprio. Azioni che con la tavoletta di plastica venivano eseguite in

Esempio d'uso

Premere E. La disposizione attuale è

1	2	3
4	5	6
7	8	9

: vengono visualizzati

momentaneamente i numeri 123 e 456, e stabilmente il numero 789. Le tre righe possono essere volendo riesaminate a rotazione premendo consecutivamente il tasto R/S. Se ora premiamo A avremo i numeri 123, 459 e 786.

000	76	Lb1	032	72	STOind	064	43	RCL	096	11	11
001	11	A	033	00	00	065	05	05	097	91	R/S
002	03	3	034	43	RCL	066	65	X	098	43	RCL
003	94	+/-	035	10	10	067	01	1	099	12	12
004	61	GTO	036	42	STO	068	00	0	100	91	R/S
005	00	0	037	00	00	069	85	+	101	43	RCL
006	23	23	038	43	RCL	070	43	RCL	102	13	13
007	76	Lb1	039	01	01	071	06	06	103	91	R/S
008	12	B	040	65	X	072	95	=	104	25	CLR
009	03	3	041	01	1	073	66	Pause	105	91	R/S
010	61	GTO	042	00	0	074	42	STO	106	61	GTO
011	00	0	043	00	0	075	12	12	107	00	0
012	23	23	044	85	+	076	43	RCL	108	95	95
013	76	Lb1	045	43	RCL	077	07	07	109	76	Lb1
014	13	C	046	02	02	078	65	X	110	15	E
015	01	1	047	65	X	079	01	1	111	09	9
016	94	+/-	048	01	1	080	00	0	112	42	STO
017	61	GTO	049	00	0	081	00	0	113	00	00
018	00	0	050	85	+	082	85	+	114	43	RCL
019	23	23	051	43	RCL	083	43	RCL	115	00	00
020	76	Lb1	052	03	03	084	08	08	116	72	STOind
021	14	D	053	95	=	085	65	X	117	00	00
022	01	1	054	66	Pause	086	01	1	118	97	DSZ
023	85	+	055	42	STO	087	00	0	119	00	0
024	43	RCL	056	11	11	088	85	+	120	01	1
025	00	00	057	43	RCL	089	43	RCL	121	14	14
026	95	=	058	04	04	090	09	09	122	09	9
027	42	STO	059	65	X	091	95	=	123	42	STO
028	10	10	060	01	1	092	42	STO	124	00	00
029	09	9	061	00	0	093	13	13	125	61	GTO
030	63	Excind	062	00	0	094	91	R/S	126	00	0
031	10	10	063	85	+	095	43	RCL	127	38	38

Figura 1. Il listato del programma che vi permette di giocare ad una "tavola del 15" ridotta, un rompicapo molto di moda prima di essere soppiantato dal magico Rubik-cubo.

modo rapido e meccanico, diventano qui più meditative ed assumono un diverso significato in dipendenza del diverso modo di vedere lo spostamento di un quadratino.

Prima di iniziare il gioco il tasto E di inizializzazione provvede a stabilire l'ordine originale dei nove numeri caricandone il valore in memoria, e la relativa posizione dello "spazio vuoto".

è in edicola
VIDEO
GIOCHI
una rivista
da non perdere!

Tre programmi di fisica e chimica con l'HP67 - HP97

di Francesco Sismondo

Calcola la quantità di radiazioni a cui sei esposto

Viviamo in un mondo radioattivo. Buona parte di queste radiazioni sono di origine naturale; alcune, comunque, sono prodotte dall'attività umana. Una unità di misura molto usata per entrambi i tipi di radiazione è il millirem (mrem) ed una tipica sorgente naturale (la radiazione di fondo) può irradiarvi con circa 100 mrem all'anno, quantità che può variare con le zone in cui venite a trovarvi.

Oltre a questo anche altri fattori concorrono a determinare la quantità annua di radiazioni cui mediamente siete esposti, come il cibo, ciò che si beve o ciò che si respira.

Sebbene scienziati e dottori non conoscano esattamente gli effetti delle diverse dosi di radiazioni a cui si può essere esposti, il Consiglio Nazionale USA di Protezione e Misura per le Radiazioni (NCRP) ha fissato a 500 mrem per anno la quantità massima di radiazioni tollerabili senza pericolo.

Varie agenzie federali americane, come la EPA e NCR, hanno stabilito limiti aggiuntivi per quel che riguarda l'apporto di radiazioni provenienti da attività umane. Gli standard EPA tollerano una dose annua di 25 mrem al di sopra della radiazione di fondo, misura abbastanza lontana dal limite di "sicurezza" della NCRP. Gli standard della NCR parlano invece di una tolleranza di 10-15 mrem in più per anno. Il limite stabilito per un impianto ad ener-

gia nucleare è di 5 mrem per anno.

Sebbene sia difficile determinare con accuratezza il numero di mrem di radioattività che ciascuno assorbe ogni anno, il questionario ed il programma seguenti vi aiuteranno a fare una stima approssimata. ■

Tavola 1. *Altitudine delle principali città*

città	altitudine in metri
Torino	239
Aosta	583
Genova	19
Brescia	144
Milano	112
Trento	194
Padova	12
Venezia	1
Udine	113
Trieste	2
Bologna	54
Livorno	3
Pisa	4
Perugia	493
Ancona	16
Roma	20
Pescara	4
Campobasso	701
Napoli	17
Bari	5
Cosenza	238
Agrigento	230
Palermo	14
Nuoro	546

Questionario

(da compilare prima di usare il programma)

1. A quanti metri sul livello del mare vi trovate? (vedere la tavola 1) _____
2. Di che materiale è fatta la vostra casa?
legno=1 cemento=2 mattoni=3 pietra=4 _____
3. Quante ore al giorno trascorrete mediamente in casa? (ricordarsi di includere la notte) _____
4. Quante ore al giorno, mediamente, trascorrete all'aperto? _____
5. Se la somma delle risposte 3 e 4 è minore di 24, in che tipo di edificio trascorrete il resto della giornata?
legno=1 cemento=2 mattoni=3 pietra=4 _____
6. Acqua, cibo e aria aggiungono circa 25 mrem per anno. Stimate la vostra dose (da 22 a 29). _____
7. Il fallout dei test delle armi atomiche contribuisce con circa 4 mrem all'anno alla vostra quantità: stimate il valore nel vostro caso. _____
8. Quanti chilometri percorrete l'anno volando ad altitudini elevate (6000 metri)? _____
9. Portate orologi con quadrante fosforescente? SI=1 NO=0 _____
10. In media, quante ore trascorrete in una stanza dove è acceso un televisore? _____

11. Raggi X:

tipo	mrem	volte per anno
torace	150	X _____ = _____
colon	450	X _____ = _____
denti	20	X _____ = _____
gastro-int.	2000	X _____ = _____
testa	50	X _____ = _____
arti	420	X _____ = _____
c. vertebr.	250	X _____ = _____
stomaco	350	X _____ = _____
altri	???	X _____ = _____
		totale = _____

Qual è il vostro totale di mrem da raggi X? _____

12. A quale distanza vivete da un impianto ad energia nucleare?

molto vicino	= 1
a un chilometro e mezzo	= 2
a 8 chilometri	= 3
più lontano ancora	= 4

13. Quante ore al giorno trascorrete, in media, in casa o nelle immediate vicinanze?

Ora usate queste informazioni e le istruzioni del programma che segue per calcolare la vostra esposizione media alle radiazioni in mrem. Una dose media annuale va dai 150 ai 200 mrem.

ISTRUZIONI

passo	procedura	battere	display
1	Introdurre il programma		
2	partenza	A	0
3	Immettere		
	A. Altezza (metri sul l/m)	Altezza R/S	Altezza
	B. Materiale della costruzione; legno=1 cemento=2 mattoni=3 pietra=4	1, 2, 3 o 4; R/S	1, 2, 3 o 4
	C. Ore al giorno in casa	Ore R/S	Ore
	D. Ore al giorno all'aperto	Ore R/S	Ore
	E. Materiale altri edifici	1, 2, 3 o 4 R/S	1, 2, 3 o 4
	F. Mrem per acqua, cibo, aria	Mrem R/S	Mrem
	G. Mrem da fallout	Mrem R/S	Mrem
	H. Km in volo per anno	Km. R/S	Km.
	I. Orologio fosf.? SI=1 NO=0	100 R/S	100
	L. Distanza da impianti nucleari (1, 2, 3, 4)	1-4 R/S	1-4
	M. Ore in casa o vicino casa	Ore R/S	Ore
4	Per conoscere una variabile in relazione a:		
	A. Altezza	RCL 1	dose dall'altezza
	B. Materiale della casa	RCL 2	dose a casa
	C. All'aperto	RCL 4	dose all'aperto
	D. Altri edifici	RCL 5	dose in altri edifici
	E. Viaggi aerei	RCL 8	dose da aerei
	F. Orologio	RCL 9	dose da orologio
	G. Televisione	RCL A	dose per televisione
	H. Impianti nucleari	RCL C	dose da impianti
	I. Mrem totali	RCL O	dose totale
5	Per un nuovo calcolo tornare al passo 2.		

REGISTRI DEI DATI

0	mrem totali
2	Materiale della casa
3	Ore in casa
4	Pre all'aperto
5	Altri luoghi
6	Cibo, acqua, aria
7	Fallout
8	Chilometri in aereo
9	Orologio
A	Televisione
B	Raggi X
C	Impianti nucleari
D	Ore vicino casa

001	*LBLA	21 11	058	*	-35	115	ST+0	35-55 00	159	ST05	35 05
002	CLRG	16-53	059	ST04	35 04	116	GT09	22 09	160	ST+0	35-55 00
003	R/S	51	060	ST+0	35-55 00	117	*LBL2	21 02	161	RTN	24
004	ST01	35 01	061	RCL5	36 05	118	R/S	51	162	*LBLb	21 16 12
005	R/S	51	062	X#0?	16-42	119	RCL3	36 03	163	RCLC	36 15
006	ST02	35 02	063	GSB3	23 03	120	2	02	164	2	02
007	R/S	51	064	RCL6	36 06	121	.	-62	165	.	-62
008	ST03	35 03	065	ST+0	35-55 00	122	7	07	166	7	07
009	R/S	51	066	RCL7	36 07	123	8	08	167	8	08
010	ST04	35 04	067	ST+0	35-55 00	124	*	-35	168	*	-35
011	R/S	51	068	RCL8	36 08	125	ST02	35 02	169	ST05	35 05
012	ST05	35 05	069	.	-62	126	ST+0	35-55 00	170	ST+0	35-55 00
013	R/S	51	070	0	00	127	GT09	22 09	171	RTN	24
014	ST06	35 06	071	0	00	128	*LBL3	21 03	172	*LBL4	21 04
015	R/S	51	072	0	00	129	2	02	173	2	02
016	ST07	35 07	073	4	04	130	4	04	174	ST09	35 09
017	R/S	51	074	*	-35	131	RCL3	36 03	175	ST+0	35-55 00
018	ST08	35 08	075	ST08	35 08	132	-	-45	176	*LBL5	21 05
019	R/S	51	076	ST+0	35-55 00	133	RCL4	36 04	177	RCLD	36 14
020	ST09	35 09	077	RCL9	36 09	134	-	-45	178	.	-62
021	R/S	51	078	1	01	135	ST0E	35 15	179	2	02
022	ST0A	35 11	079	X=Y?	16-33	136	RCL5	36 05	180	*	-35
023	R/S	51	080	GSB4	23 04	137	1	01	181	ST0C	35 13
024	ST0E	35 12	081	RCLA	36 11	138	X=Y?	16-33	182	ST+0	35-55 00
025	R/S	51	082	.	-62	139	GT0a	22 16 11	183	GT08	22 08
026	ST0C	35 13	083	1	01	140	CLX	-51	184	*LBL6	21 06
027	R/S	51	084	5	05	141	4	04	185	RCLD	36 14
028	ST0D	35 14	085	*	-35	142	X=Y?	16-33	186	.	-62
029	RCL1	36 01	086	ST0A	35 11	143	GT0b	22 16 12	187	0	00
030	3	03	087	ST+0	35-55 00	144	RCLC	36 15	188	2	02
031	0	00	088	RCLB	36 12	145	2	02	189	*	-35
032	÷	-24	089	ST+0	35-55 00	146	.	-62	190	ST0C	35 13
033	4	04	090	RCLC	36 13	147	5	05	191	ST+0	35-55 00
034	2	02	091	1	01	148	*	-35	192	GT08	22 08
035	+	-55	092	X=Y?	16-33	149	ST05	35 05	193	*LBL7	21 07
036	ST01	35 01	093	GT05	22 05	150	ST+0	35-55 00	194	RCLD	36 14
037	ST+0	35-55 00	094	CLX	-51	151	RTN	24	195	.	-62
038	RCL2	36 02	095	2	02	152	*LBLa	21 16 11	196	0	00
039	1	01	096	X=Y?	16-33	153	RCLC	36 15	197	0	00
040	X=Y?	16-33	097	GT06	22 06	154	1	01	198	2	02
041	GT01	22 01	098	CLX	-51	155	.	-62	199	*	-35
042	CLX	-51	099	3	03	156	9	09	200	ST0C	35 13
043	4	04	100	X=Y?	16-33	157	4	04	201	ST+0	35-55 00
044	X=Y?	16-33	101	GT07	22 07	158	*	-35	202	GT08	22 08
045	GT02	22 02	102	0	00				203	R/S	51
046	2	02	103	ST0C	35 13						
047	.	-62	104	*LBL8	21 08						
048	5	05	105	RCL0	36 00						
049	RCL3	36 03	106	R/S	51						
050	*	-35	107	*LBL1	21 01						
051	ST02	35 02	108	RCL3	36 03						
052	ST+0	35-55 00	109	1	01						
053	*LBL9	21 09	110	.	-62						
054	RCL4	36 04	111	9	09						
055	2	02	112	4	04						
056	.	-62	113	*	-35						
057	5	05	114	ST02	35 02						

Dov'è il vostro orizzonte

Immaginate di essere con un vostro amico sulla riva di un grande lago. Guardando lontano sull'acqua vedete l'orizzonte, il punto in cui il cielo e l'acqua paiono congiungersi. L'orizzonte che voi vedete si trova alla stessa distanza da voi di quello che vede il vostro amico? Può essere, o meno. Come indicato dalla figura 1 dipende dall'altezza dei vostri occhi dal terreno.

Una persona i cui occhi sono nel punto h_2 può vedere un orizzonte, b_2 , ad una distanza maggiore che non una persona i cui occhi sono nel punto h_1 .

Possiamo essere più specifici?

Sì, dato che il teorema di Pitagora dice che l'ipotenusa di un triangolo rettangolo è uguale alla radice quadrata della somma dei quadrati costruiti sugli altri due lati ($c^2 = a^2 + b^2$), e quindi:

$$\begin{aligned} r^2 + d^2 &= (r+h)^2 \\ d^2 &= (r+h)^2 - r^2 \\ d &= \sqrt{(r+h)^2 - r^2} \end{aligned}$$

dove h è l'altezza degli occhi della persona dal livello del suolo, r il raggio della terra e d la distanza della persona dall'orizzonte.

Quindi, assumendo un raggio di 6378 chilometri, la distanza dell'orizzonte per una persona i cui occhi distano dal suolo metri 1,60 sarà di km. 4.52. Per una persona i cui occhi distano dal suolo m. 1,80, la distanza dell'orizzonte sarà km 4.79: in questo esempio un incremento di 20 cm. in h dà un incremento di 270 metri nella distanza dell'orizzonte.

Questo è il motivo per cui l'antenna radar dei piccoli battelli è situata alcuni metri più in alto del ponte. Questo è inoltre il principio che aiuta a determinare l'altitudine necessaria per un satellite di comunicazioni che deve ricevere un segnale da un punto della terra e ritrasmetterlo in un altro.

Il seguente programma calcola d per ogni valore di h . È anche possibile scegliere un valore con cui incre-

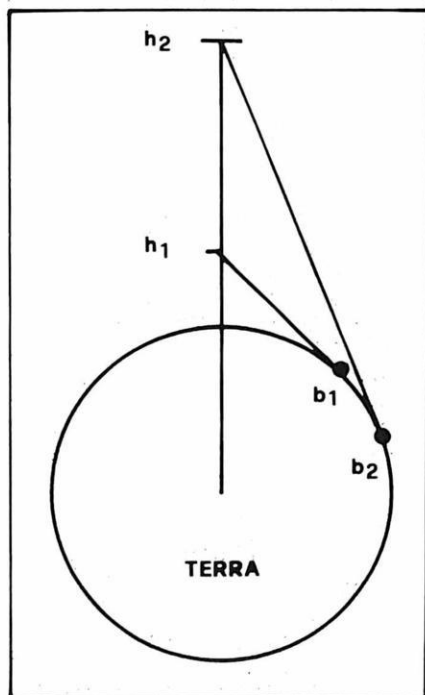


Figura 1. Posizione dell'orizzonte da due diverse altezze

mentare h e il programma visualizzerà ripetutamente la nuova h con il suo valore d associato.

Modifiche

Siccome la distanza dell'orizzonte è in parte determinata dal raggio del pianeta su cui ci si trova, è necessario apportare delle modifiche al pro-

gramma se si vogliono simulare queste eventualità su di un altro pianeta.

Si tratta semplicemente di cambiare il valore per il raggio del pianeta alle locazioni del programma 002-

```

001 *LBLA      21 11
002 ST01      35 01
003          6   06
004          3   03
005          7   07
006          8   08
007 ST03      35 03
008 R/S       51
009 ST02      35 02
010 *LBL1     21 01
011          1   01
012          0   00
013          0   00
014          0   00
015          ÷   -24
016 RCL3      36 03
017          +   -55
018          X²   53
019 RCL3      36 03
020          X²   53
021          -   -45
022          √X   54
023 ST05      35 05
024 RCL2      36 02
025 PRTX      -14
026 RCL5      36 05
027 PRTX      -14
028 RCL1      36 01
029 ST+2     35-55 02
030 RCL2      36 02
031 GT01     22 01
032 R/S       51
    
```

Pianeta/satellite	Raggio in chilometri
Mercurio	2400
Venere	6100
Terra	6378
Luna	1738
Marte	3400
Fobo	5
Urano	24850
Titania	350
Miranda	175
Asteroidi:	
Cerere	382
Icaro	0.8
Vesta	189

ISTRUZIONI PER L'UTENTE

Passo	Procedura	Battere	Display	Registri dei dati
1	Introdurre il programma		0	1 incremento in metri
2	Immettere l'incremento in metri; questo è il valore di cui è previsto debba essere incrementata l'altezza.	Incremento R/S		2 altezza in metri 3 raggio del pianeta o satellite
3	Immettere l'altezza h in metri (distanza degli occhi dal livello del suolo). La stampante continua a stampare la nuova altezza con associata la distanza dell'orizzonte.	h R/S	Altezza in metri	5 distanza dell'orizzonte in chilometri
4	Per fermare il programma premere R/S.	Distanza in Km.		
5	Per ripartire, tornare al passo 2.			

005. Per raggi di lunghezza minore di quattro cifre uno o più passi di programma devono essere cancellati, e per raggi maggiori di quattro cifre

altri ne devono essere inseriti.

La tavola fornisce i raggi approssimati per diversi pianeti e per i loro satelliti naturali.

Determinazione di acidi e basi

Acidi e basi sono sostanze comuni in chimica, e molto spesso devono essere identificate e quantificate.

Gli acidi hanno un sapore acido e rendono blu la carta di tornasole rossa, mentre le basi fanno diventare rossa quella blu. Ad ogni modo, è la quantità di ioni idrogeno presenti che identifica una soluzione come acida o basica.

L'acqua, quando è neutra, contiene 1.00310^{-8} moli di H_3O^+ per litro. Se la concentrazione di H_3O^+ è maggiore di quella dell'acqua pura, la soluzione è acida. Al contrario, le soluzioni che contengono meno di 1×10^{-7} moli di H_3O^+ per litro sono basiche. Questo sistema per quantificare acidi e basi è molto accurato ma anche molto complicato, e così molti chimici usano spesso un metodo alternativo conosciuto come scala pH.

La scala pH è stata sviluppata

come una maniera relativamente semplice per esprimere l'acidità o l'alcalinità di una soluzione. La scala è logaritmica, e il suo intervallo

va da 0 a 14. Le soluzioni acide hanno un pH minore di 7, mentre quelle basiche hanno un pH maggiore di 7. Un pH di 7 indica che la soluzione è neutra.

Il pH è definito come segue:

$$pH = \log \frac{1}{H_3O^+}$$

Se pH è noto, H_3O^+ può essere calcolato in questo modo:

$$H_3O^+ = \text{antilog}(-pH)$$

Questo programma incrementa e decrementa di continuo la concentrazione di H_3O^+ e mostra il pH risultante all'interno dei parametri da voi scelti. Sono possibili anche singole determinazioni di pH per date concentrazioni di H_3O^+ .

Una semplice simulazione

La concentrazione di H_3O^+ di una data soluzione è di 1×10^{-7} moli per litro. Se la concentrazione viene aumentata con incrementi di $0,4 \times 10^{-5}$ moli per litro fino a che viene superata una concentrazione finale di 1×10^{-5} moli per litro, quale sarà il valore di pH associato ad ogni nuova soluzione di H_3O^+ minore di 1×10^{-5} .

Dopo l'immissione del program-

ISTRUZIONI PER L'UTENTE

Passo	Procedura	Battere	Display
1	Immettere il programma		
2	Partenza	A	
3	Immettere i dati		
	A. Concentrazione iniziale, C_I di H_3O^+	C_I , R/S	C_F
	B. Concentrazione finale, C_F di H_3O^+	C_F , R/S	C_F
	C. Incremento, S	S, R/S	S
			Concentrazione di H_3O^+ con il valore pH associato
5	Per un'altra simulazione, tornare al passo 2		

REGISTRI DEI DATI

0	Concentrazione iniziale e corrente
1	Incremento
2	Concentrazione finale

è in edicola il nuovo numero



RIVER RAID IL NUOVO GIOCO ACTIVISION
TUTTI CON NOI ALLA SUPERGARA DEL SIM!
CINEMA E VIDEOGAMES A BRACCETTO
IN PROVA TUTTE LE NOVITA' ESTIVE
COMPUTER: PROGRAMMI, PROGRAMMI



UNA PUBBLICAZIONE DEL
GRUPPO EDITORIALE JACKSON

001	*LBLA
002	DEP2
003	R/S
004	ST00
005	R/S
006	ST02
007	R/S
008	ST01
009	RCL0
010	PRTX
011	*LBL1
012	1/X
013	LOG
014	PRTX
015	RCL1
016	ST+0
017	X<0?
018	GT08
019	RCL0
020	RCL2
021	*LBL2
022	X<Y?
023	GT0C
024	RCL0
025	PRTX
026	GT01
027	*LBLC
028	RCL2
029	R/S
030	*LBLB
031	RCL2
032	RCL0
033	GT02
034	R/S

ma, memorizzate la concentrazione iniziale, 1×10^{-7} , nel registro 00 come segue: 1EE7+/-ST000. Allo stesso modo memorizzate la concentrazione finale, 1×10^{-5} , nel registro 02 e l'incremento, $0,4 \times 10^{-5}$ nel registro 01. Premete R/S e la concentrazione iniziale sarà brevemente mostrata seguita dal valore di pH ad essa associato.

In questo caso:

H_3O^+	pH
$1,00 \times 10^{-7}$	7,00
$4,10 \times 10^{-6}$	5,39
$8,10 \times 10^{-6}$	5,09

La compressione dei dati

di Paolo Capobussi

Il numero delle memorie disponibili in una calcolatrice programmabile, quale può essere la anziana TI 59, è spesso soggetto a limitazioni notevoli. Purtroppo, infatti, le programmabili di moderna concezione hanno in gran parte adottato il metodo di ripartizione della memoria in area dati e area programmi, con automatica o selezionabile suddivisione dell'ampiezza relativa.

Questo è un concetto universalmente adottato, in genere in tutti i microcomputer, giacché è relativamente semplice spostare dei puntatori che, da sistema operativo, delimitino le due aree tenendo conto dell'occupazione richiesta.

Quando però la disponibilità di memoria è scarsa, e nelle programmabili questa carenza è certamente sensibile, capita di creare programmi particolarmente efficaci e complessi, ma che non possono girare che con pochi o addirittura pochissimi dati. A volte succede perfino che una non accorta previsione renda impossibile l'utilizzo del programma in quanto la macchina non è più in grado di supportare anche una sufficiente area di dati.

Il problema è particolarmente sentito nel caso dei programmi statistici o di analisi di tendenza e nei problemi riguardanti l'astronomia, ove devono essere analizzate grosse quantità di dati o venga richiesto un numero notevole di costanti.

Nel caso di una calcolatrice come la TI 59 o altre di simile concezione, operanti in linguaggi macchina specializzati, le memorie in questione

vengono normalmente chiamate "registri". Sono, quindi, l'equivalente logico delle variabili nei linguaggi ad alto livello, come il Basic già adottato da molti altri pocket. In ciascuno di questi registri è possibile memorizzare un numero di tante cifre quante sono quelle che la macchina è in grado di supportare, equivalenti a quelle visibili sul display.

Quindi, normalmente, un solo numero per ciascun registro. Volendo memorizzare due numeri in ogni celletta di memoria, molto spesso la tecnica adottata è quella di accoppiarli rendendone uno la parte decimale dell'altro. Naturalmente si dimezza la quantità di cifre costituenti ogni numero, e inoltre si hanno problemi di gestione dei numeri negativi, ma può diventare un ottimo metodo se si pensa che generalmente si lavora con numeri decisamente più piccoli delle usuali dieci o tredici cifre che la calcolatrice è in grado di gestire. Il programma che permette di realizzare questo tipo di impacchettamento è semplice e intuitivo: diviso uno dei due numeri per una potenza di dieci pari alla quantità delle sue cifre intere, lo si somma all'altro reso intero. Si ha così un numero formato da una parte intera e da una decimale, facilmente separabili tramite le funzioni INTeger e FRACtionary.

Una semplice operazione di moltiplicazione, inversa alla precedente divisione, restituirà anche il primo numero nella sua forma originale. Per lunghe tabelle di numeri interi non molto grandi questa tecnica permette di raddoppiare, quindi, il

numero dei registri disponibili. Nel caso servissero più dati ancora, o la capacità di memoria riservata ai dati fosse stata drasticamente ridotta da un programma eccessivamente lungo, esistono altre tecniche di impacchettamento. Tali tecniche richiamano spesso a dei concetti piuttosto interessanti.

Le caratteristiche

Innanzitutto il metodo che presentiamo riesce ad impaccare dati di lunghezza qualsiasi in numero particolarmente elevato: fino a 43 numeri in solo registro di una TI 59.

L'unica limitazione è la differenza assoluta esistente tra il numero a valore massimo e quello a valore minimo.

La possibilità di condensazione dipende, infatti, proprio da questo "range". Il valore riportato si riferisce ad un range di 1. Sarà possibile ad esempio, memorizzare quarantatre numeri come 8720 e 8721, o anche più elevati, purché differenti per una sola unità.

All'aumentare della differenza assoluta tra numero massimo e minimo diminuisce la densità possibile. Comunque per una differenza di 10 è ancora possibile condensare in un solo registro una sequenza di 12 numeri! Ricordiamoci sempre che la lunghezza non ha importanza: in quest'ultimo caso potrebbero essere le prime dieci cifre decimali o dei numeri compresi tra, ad esempio, 54320 e 54330.

000	76	LBL	033	10	E'	066	74	SM*	099	55	÷
001	11	A	034	43	RCL	067	01	01	100	43	RCL
002	47	CMS	035	03	03	068	97	DSZ	101	05	05
003	25	CLR	036	65	×	069	00	00	102	95	=
004	08	8	037	97	DSZ	070	17	B'	103	59	INT
005	42	STD	038	00	00	071	69	DP	104	65	×
006	01	01	039	10	E'	072	21	21	105	32	X↑T
007	01	1	040	01	1	073	61	GTD	106	43	RCL
008	03	3	041	95	=	074	16	A'	107	05	05
009	55	÷	042	42	STD	075	76	LBL	108	95	=
010	53	(	043	04	04	076	12	B	109	22	INV
011	91	R/S	044	76	LBL	077	25	CLR	110	44	SUM
012	42	STD	045	16	A'	078	08	8	111	07	07
013	06	06	046	43	RCL	079	42	STD	112	32	X↑T
014	75	-	047	02	02	080	01	01	113	85	+
015	91	R/S	048	42	STD	081	76	LBL	114	43	RCL
016	75	-	049	00	00	082	18	C'	115	06	06
017	01	1	050	89	↑	083	43	RCL	116	95	=
018	54	)	051	61	GTD	084	04	04	117	99	PRT
019	94	+/-	052	89	↑	085	42	STD	118	43	RCL
020	42	STD	053	76	LBL	086	05	05	119	03	03
021	03	03	054	17	B'	087	43	RCL	120	22	INV
022	28	LOG	055	43	RCL	088	02	02	121	49	PRD
023	95	=	056	03	03	089	42	STD	122	05	05
024	59	INT	057	64	PD*	090	00	00	123	97	DSZ
025	66	PAU	058	01	01	091	73	RC*	124	00	00
026	42	STD	059	76	LBL	092	01	01	125	19	D'
027	02	02	060	89	↑	093	42	STD	126	69	DP
028	42	STD	061	91	R/S	094	07	07	127	21	21
029	00	00	062	75	-	095	76	LBL	128	01	1
030	69	DP	063	43	RCL	096	19	D'	129	61	GTD
031	30	30	064	06	06	097	43	RCL	130	18	C'
032	76	LBL	065	95	=	098	07	07	131	00	0

Figura 1. Il programma PACK in linguaggio adatto alla Texas Instruments TI 59.

Sono sempre richiesti, naturalmente, solo numeri interi. Se questa è una limitazione per calcoli scientifici, molto spesso ciò non diventa ostacolo. Sono moltissime le applicazioni in cui delle tabelle vengono costruite con numeri ripetitivi, interi e quindi di facile maneggiabilità da parte del programma che presentiamo.

Nulla vieta, inoltre, di abbinare a tale programma una ulteriore routine capace di rendere interi dei numeri con cifre decimali, per poi ricostruirli dopo opportuno disimpaccamento.

La precisione è garantita dal pro-

cedimento matematico che tale programma segue.

Il metodo

Di tale procedimento è utile osservarne gli aspetti principali. Innanzitutto il principio. Si considerano tutti i numeri introdotti come numeri in base B, ove con B si intende una base numerica definitiva come numero maggiore di 1 del range, ovvero della differenza in valore assoluto tra numero più grande e numero più piccolo tra quelli introdotti.

Supponiamo, ad esempio, che i nostri numeri da condensare siano i seguenti: 5, 0, 3, 5. Il più piccolo è 0, il più grande è 5.

Il range è, quindi, 5. Dal programma verrà considerato un numero costituito dai singoli numeri introdotti, che nel nostro esempio coincidono con le quattro cifre dette, considerate quale numero in base B.

Nel nostro caso, quindi, il numero viene gestito come 5035_B. Se trasformiamo tale numero nell'equivalente decimale otteniamo un altro numero in base 10 che possiede una cifra in meno del numero dato: da questo il

$$P = a_0 B^n + a_1 B^{n-1} + \dots + a_n B_0$$

$$a_i = (V_i - m) \quad i = 0 \dots n$$

P : polinomio di grado n
 a_i : coefficienti del polinomio
 V_i : valori introdotti compresi nel range
 m : valore minimo tra quelli considerati
 n : numero dei valori da impaccare

Figura 2. Il polinomio che viene costruito dal programma per poter condensare in un solo registro fino a 43 valori differenti, densità variabile secondo il range dei valori da impaccare, ma indipendente dal numero di cifre di ciascun valore.

risparmio di spazio.

In realtà tutto questo è realizzato da programma costruendo un polinomio di grado pari al numero di dati da impaccare, i cui coefficienti sono i valori introdotti a cui viene sottratto il valore più piccolo, poi risommato al momento della restituzione dei numeri.

Quindi il programma esegue le seguenti funzioni:

- a) dato il numero più piccolo e quello più grande tra quelli da memorizzare, calcola il numero massimo di dati impaccabili in un solo registro (passi da 000 a 029);
- b) calcola la base che costituisce la variabile X del polinomio di impaccamento, successivamente costruito (sempre nei passi da 010 a 021);
- c) calcola (passi da 032 a 043) il valore che utilizzerà come divisore del polinomio stesso per estrarne i singoli coefficienti, nel procedimento di disimpaccamento realizzato in una opportuna routine. Poi si ferma in attesa dei dati;
- d) ad ogni dato introdotto, compreso tra il massimo e il minimo precedentemente individuati, sottrae il valore minimo e considera tale nuovo valore come

coefficiente di un polinomio in B , essendo B la base ricavata come precedentemente visto. Costruisce così tale polinomio ottenendone un valore che memorizza in un registro, a partire dall'ottavo.

Quando in tale registro non sono inseribili più cifre, ovvero il polinomio ha raggiunto il valore massimo consentito, automaticamente il procedimento inizia nuovamente per un registro successivo, e così via fino ad esaurimento dei dati (passi da 053 a 074);

- e) il procedimento di disimpaccamento è alquanto più semplice. La routine che inizia alla etichetta B , infatti, recupera il polinomio, lo divide opportunamente così da isolare una parte intera costituita esattamente dal coefficiente interessato (da passi 097 a 103). Somma a tale coefficiente il numero più piccolo che in fase di impaccamento aveva sottratto e ottiene così nuovamente il numero originale (passi da 112 a 117).

Il metodo è interessante non solamente, riteniamo, per la calcolatrice programmabile TI 59 ove è stato implementato. Poter impaccare dei

dati è spesso necessità per risparmio della memoria, laddove consistenti programmi mettono alla prova anche il più stringato programmatore.

Certamente non è un metodo che aggiunge chiarezza ad una gestione di dati, ma una volta accertatisi del buon funzionamento può essere trattato come routine a se stante richiamabile ogni qualvolta se ne presenti necessità. Supplirà la documentazione allegata a questa povertà autoesplicativa del programma stesso.

Per utilizzare il programma:

- 1) Premere A per avviare la routine di inizializzazione.
- 2) Inserire il valore più **piccolo** tra quelli da memorizzare e premere R/S .
- 3) Inserire il valore più **grande** e premere R/S .

Il programma visualizza per un istante il numero massimo di valori che possono essere memorizzati in ciascun registro a partire dal numero 08.

Appare in ultimo il valore di π -greco (3,1415...) che indica il termine di ciascuna fase (facilmente distinguibile dai valori introducibili in quanto π -greco è decimale).

- 4) Inserire i valori da memorizzare, in qualunque ordine, purché compresi tra il valore massimo e quello minimo.
Dopo ciascun valore premere R/S .

All'apparire di π -greco possiamo proseguire l'introduzione che andrà ad interessare un registro successivo.

- 5) Se tutti i numeri sono stati introdotti, ma il registro non è saturo e quindi non appare π -greco, introdurre altri numeri qualunque, purché nel range previsto, fino all'apparire di 3,1415...

Gli errori di introduzione sono rimediabili eseguendo la sequenza di tasti 2nd Op 31 0 STO 2nd Ind 01, solamente dopo l'apparizione di π -greco. Fatto ciò il registro andrà ricaricato di nuovo. Per ottenere la restituzione dei dati originali premere B .

I numeri vengono stampati o, sostituendo il passo 117 con un R/S , semplicemente visualizzati. ■

Master Mind su HP-29C

di Enrico Bossaglia

La stesura di un programma in Basic che consenta di giocare a Master Mind con un personal computer (affidando al computer il ruolo di Codificatore) non presenta particolari difficoltà; anzi, direi che si tratta di un lavoro addirittura banale. Se però cerchiamo di risolvere lo stesso problema utilizzando una piccola calcolatrice programmabile, per esempio la HP-29C, che lavora in RPN con un set molto limitato di istruzioni e con una piccola memoria di programma, ci accorgiamo che la cosa diventa più interessante.

Quando infatti mi cimentai per la prima volta nella stesura di questo programma (rispondendo all'amichevole sfida di un amico), esso risultò troppo lungo per la memoria di 98 passi disponibile sulla macchina. Iniziai così una serie di tentativi per ridurne le dimensioni giungendo infine al programma qui riportato.

Per coloro, pochi, che ancora non conoscono questo gioco, elenchiamo le principali regole:

Il M.M. si gioca tra due giocatori, uno funge da CODIFICATORE e l'altro da DECIFRATORE.

All'inizio di una partita il Codificatore compone una sequenza di quattro cifre (non necessariamente tutte diverse tra loro) detta CODICE SEGRETO (C.S.).

Le cifre possono essere scelte tra le sei seguenti: 0, 1, 2, 3, 4, 5.

Il Decifratore deve cercare di scoprire il C.S. in base alle risposte che il Codificatore dà ai suoi tentativi di decifrazione.

Il Codificatore deve fornire le ri-

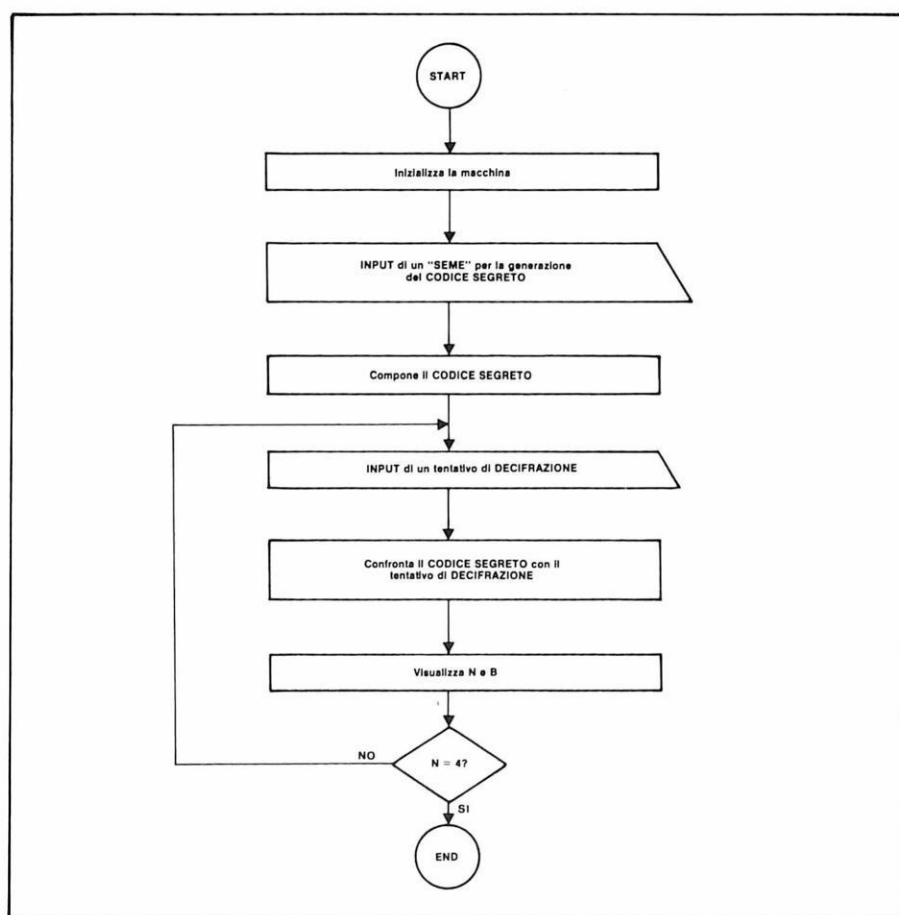


Figura 1. Il diagramma di flusso semplificato di questo classico Master Mind, redatto per i pochi passi disponibili di una HP-29C.

sposte nel modo seguente:

- Dichiarare quante sono le cifre giuste al posto giusto (N)
- Dichiarare quante sono le cifre giuste al posto sbagliato (B).

È ovvio che se una cifra ha già

fruttato un punteggio non può essere riutilizzata nel corso dello stesso tentativo.

Per esempio:

CODICE SEGRETO
TENTATIVO DI DECIFRAZIONE
RISPOSTA DEL CODIFICATORE

1442	1452	0313
1434	1444	2355

N=2 B=1 N=2 B=0 N=1 B=0

Nel nostro caso il calcolatore gioca da Codificatore.

Il programma in oggetto deve quindi svolgere le operazioni indicate nel diagramma di flusso di fig. 1, ossia generare, previa introduzione di un seme numerico, una sequenza di quattro cifre comprese tra 0 e 5, quindi confrontare questo C.S. con i tentativi di decifrazione assegnando di volta in volta la risposta appropriata. Al fine di rendere ragione di certe scelte di programmazione è riportata in Appendice B una sintesi

delle principali caratteristiche della calcolatrice programmabile HP-29C.

Come si può vedere non esistono né "flag" direttamente interrogabili né un generatore di numeri casuali.

In questo programma vengono usati tutti i 98 passi disponibili, tutte le 10 label e tutti i 16 registri primari.

Prima di entrare nel dettaglio del programma vediamo come sono stati impiegati i 16 registri di memoria primari.

R0	: Viene usato come Contatore e come Puntatore per l'indirizzamento indiretto.
R1, R2, R3, R4	: Ospitano rispettivamente la prima, la seconda, terza, e quarta cifra del C.S. (indicato con A, B, C, D).
R5	: Viene utilizzato sia come totalizzatore N, sia come registro ausiliario.
R6	: Viene utilizzato come totalizzatore B.
R7	: Contiene la costante 1.7 utilizzata nella generazione del C.S.
R8	: Viene artificiosamente utilizzato come contatore principale (lento) nella seconda parte del prog.
R9	: Contiene il Puntatore (14) della quarta cifra del tentativo di decifrazione.
R.0	: Contiene la differenza (10) tra l'indirizzo di una cifra del C.S. e quello della corrispondente cifra del tentativo di decifrazione.
R.1, R.2, R.3, R.4	: Ospitano rispettivamente la prima, seconda, terza e quarta cifra del tentativo di decifrazione (indicate con a, b, c, d).
R.5	: Contiene il generatore corrente del C.S.

Funzionamento del programma

È consigliabile seguire la descrizione del programma sul diagramma di flusso di fig. 2.

Dopo aver caricato e memorizzato le costanti (come indicato in fig. 3) si può iniziare il gioco digitando un seme numerico intero o decimale compreso tra 1 e 10.

Al comando GSB 7 il programma genera il C.S. nel seguente modo:

- Inizializza R0 memorizzandovi il Puntatore (4) dell'ultima cifra del C.S.,
- Somma il Seme a R.5,
- Richiama R.5 e divide per Pi (Pi greco),
- Memorizza in R8 la parte frazionaria (sicuramente composta da

cifre non tutte nulle),

- Attua un ciclo (LBL O) che estrae da R8 numeri decimali minori di 10, effettua la divisione per R7 e ne prende la parte intera (sicuramente minore o uguale a 5), la quale viene memorizzata per via indiretta nel registro corrente tra i quattro dedicati al C.S.,
- Finisce il ciclo mediante l'istruzione DSZ (Decrement and Skip if Zero),
- Inizializza R0 memorizzandovi il Puntatore (14) dell'ultima cifra del tentativo di decifrazione,
- Ferma il programma e attende l'input delle cifre costituenti il tentativo di decifrazione.

Introdotta la cifra di tentativo secondo le modalità indicate dalle Istruzioni per l'Utente, il programma prosegue con il comando GSB 1.

La sequenza di operazioni svolte dal calcolatore è la seguente:

- Effettua un ciclo iterativo (LBL 1) per la memorizzazione indiretta di ogni cifra del tentativo di decifrazione nel registro corrente (R.4, R.3 etc.); il metodo di memorizzazione usato consente di uscire dal ciclo con l'uso di un numero minimo di istruzioni e senza modificare i registri non interessati alla memorizzazione del tentativo di decifrazione.
- Azzeri i registri R5 ed R6 (rispettivamente totalizzatori N e B).
- Inizializza R0 memorizzandovi l'indirizzo (4) dell'ultima cifra del C.S.
- Effettua un ciclo iterativo (LBL 2) per la ricerca delle cifre giuste al posto giusto (totalizzatore N); R0 è utilizzato come "offset" per l'indirizzamento delle cifre di tentativo.
- Dopo ogni confronto, se le cifre sono disuguali, decrementa R0 (contatore) e ricomincia l'iterazione, se invece le cifre sono uguali, aggiunge 1 al totalizzatore R5 (N) quindi somma 10 ai registri contenenti le cifre in questione (questa operazione consente l'esclusione da successive assegnazioni di punteggi delle cifre in oggetto).
- Esaurite le cifre da confrontare l'istruzione DSZ permette l'uscita dal ciclo.
- Il programma si ferma per consentire la visualizzazione di R5 (N) (ciò è necessario in quanto nel seguito del programma R5 viene riutilizzato e quindi il totalizzatore N andrebbe perduto).
Mediante la pressione del tasto R/S si fa quindi ripartire il programma che intraprende la ricerca delle cifre giuste al posto sbagliato:
- Inizializza R8 con il numero 4 (R8 avrà ora la duplice funzione di puntatore della cifra corrente del C.S. e di contatore principale).
- Inizia l'esecuzione del ciclo principale (LBL 4).
- Memorizza il contenuto del registro X, che inizialmente è 4, in R0.
- Richiama indirettamente la cifra corrente del C.S. e la memorizza in R5 (registro ausiliario).
- Inizializza R0 con 14 (indirizzo

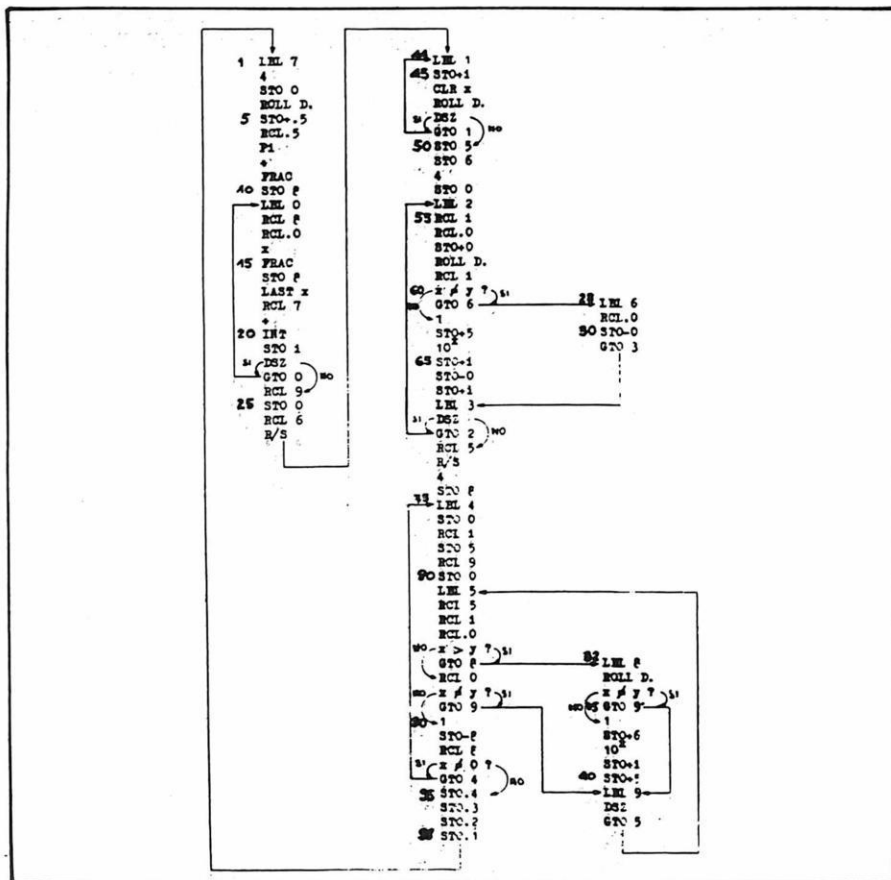


Figura 2. Nel dettaglio, un diagramma che ci permetterà di seguire meglio il listato stesso del programma.

Figura 3. Le istruzioni per l'utente, peraltro rispecchianti la semplicità d'uso della macchina in questione.

ISTRUZIONI	INPUT DATI UNITÀ	TASTI	OUTPUT DATI UNITÀ
1. Caricare il programma			
2. Memorizzare le costanti	1.7 14 10	STO 7 STO 9 STO.0	
3. Generazione automatica del codice segreto A B C D	SEME*	GSB 7	
4. Tentativo di decifrazione con la sequenza di cifre a b c d	a b c d	ENTER ENTER ENTER GSB 1 R/S	N B
5. Per un nuovo tentativo di decifrazione tornare al punto 4.			
6. Per la generazione automatica di un nuovo codice segreto tornare al punto 3.			
* Il seme può essere un qualunque numero intero o decimale compreso tra 1 e 10.			

dell'ultima cifra del dentativo di decifrazione).

- Inizia un ciclo nidificato (LBL 5) per il confronto delle quattro cifre del tentativo di decifrazione con la cifra corrente del C.S. memorizzata in R5.

La procedura di confronto relativa a quest'ultimo ciclo (LBL 5), è attuata richiamando nella Catasta Operativa rispettivamente R5 (C.S.), Ri (i = 14, 13, 13, 11: tentativo), R.0 (costante 10).

La Catasta Operativa si presenta così:

T C.S.
Z R5 =
Y Ri = Tentativo di decifrazione
X R.0 = 10

Per prima cosa vengono confrontati i registri X e Y (passo 85) per verificare una precedente assegnazione di punteggio; in caso affermativo si esegue un test su R0 (puntatore) per vedere se ci sono altre cifre da confrontare con R5 (cifra corrente del C.S.) e se anche in questo caso la risposta è sì, R0 viene decrementato e nuovamente intrapreso il ciclo LBL 5.

Se invece il "Loop" nidificato è terminato viene decrementato R8 (contatore principale e puntatore) e portato sul registro X per controllare se restano altre cifre del C.S. da utilizzare nei confronti. Se ciò si verifica viene ripetuto il ciclo LBL 4, altrimenti il programma riprende dal passo 95.

Prima di proseguire esaminiamo cosa succede quando l'esito del test per l'accertamento di avvenuta assegnazione punteggio (passo 85) è negativo.

Al passo 86 si ha un'istruzione di salto alla LBL 8, quindi la Catasta Operativa viene fatta ruotare verso il basso ottenendo la seguente configurazione:

T R.0 = 10
Z
Y R5 = C.S.
X Ri = Tentativo di decifrazione

rendendo possibile il confronto fra i registri X e Y.

Se i numeri sono uguali viene ag-

Principali Caratteristiche della Calcolatrice Programmabile HP-29C

- Programmazione nel sistema logico RPN.
- 98 Passi di programma disponibili.
- Catasta Operativa Automatica composta da quattro registri (X, Y, Z, T) di cui X visualizzato, manipolabile con le istruzioni ROL DOWN e $\times \div$.
- 16 Registri Primari ad Indirizzamento Diretto o Indiretto (indirizzo contenuto nel registro R0).

REGISTRO	INDIRIZZO DIRETTO	INDIRIZZO INDIRETTO
R0	0	0
R1	1	1
R2	2	2
R3	3	3
R4	4	4
R5	5	5
R6	6	6
R7	7	7
R8	8	8
R9	9	9
R.0	.0	10
R.1	.1	11
R.2	.2	12
R.3	.3	13
R.4	.4	14
R.5	.5	15

Esempio di Indirizzamento Diretto:

225 STO 5 memorizza 225 nel registro R5

32 STO .4 memorizza 32 nel registro R.4

Esempio di Indirizzamento Indiretto:

Se R0 contiene 5, 225 STOi memorizza 225 nel registro R5

Se R0 contiene 14, 32 STOi memorizza 32 nel registro R.4

- Registro R0 utilizzabile anche come Contatore (istruz. ISZ, DSZ).
- Operazioni aritmetiche nei registri di memoria STO+, STO-, STO $\times$, STO $\div$ con indirizzamento diretto e indiretto.
- 2 Istruzioni di salto: GTO (GO TO) e GSB (GOSUB) seguite da una label numerica.
- 10 Label numeriche.
- 8 Test comparativi: 4 tra i registri X e Y e 4 tra il registro X e il numero 0.
- Istruzione R/S (RUN/STOP) per effettuare Input dati o per fermare o riavviare il programma.

giunto 1 ad R6 (totalizzatore B). Anche in questo caso viene sommato ad R5 ed Ri il numero 10 per impedire successive assegnazioni di punteggio.

Se invece i due numeri sono diversi viene semplicemente decrementato R0 (puntatore cifre di tentativo) e il programma riprende dalla LBL 5.

Ora resta soltanto lo spazio per quattro istruzioni mediante le quali il programma deve essere in grado di visualizzare R6 (totalizzatore B) ed inizializzare il calcolatore per l'esame di un nuovo tentativo di deci-

frazione. Per fare ciò è sufficiente inserire in questi quattro passi di programma disponibili le istruzioni di azzeramento dei quattro registri che ospitano il tentativo di decifrazione. Quando infatti il calcolatore esegue lo "skip" dell'istruzione 94, contiene 0 nel registro X. Questo fatto consente di azzerare molto facilmente i registri ospiti delle cifre di tentativo. Poiché, arrivato al passo 98 (l'ultimo), il calcolatore non trova un'istruzione di Stop, continua l'elaborazione ricominciando dal passo 1. Viene quindi rieseguita la subroutine che genera le cifre del

RIVISTE JACKSON.
LA VOCE
PIÙ AUTOREVOLE
NEL CAMPO
DELL'ELETTRONICA
E DELL'INFORMATICA.

l'Elettronica
PERSONAL
SOFTWARE
AUTOMAZIONE

strumenti
MUSICALI

INFORMATICA

elektor
Bit

VIDEO
Giochi
elettronica
OGGI



GRUPPO EDITORIALE
JACKSON

LISTATO DEL PROGRAMMA					
02 LBL 7	Generatore del codice segreto ABCD	34 $x \neq$	Incremento totalizzatore B e attiva- zione "Flag"	67 STO+i	Visualizz. totalizz. N.
03 4		35 GTO 9		68 LBL 3	
04 STO 0		36 1		69 DSZ	
05 ROLL D.		37 STO+6		70 GTO2	
06 STO+.5		38 10^x		71 RCL 5	
07 RCL. 5		39 STO+i		72 R/S	
08 PI		40 STO+5		73 4	
09 ÷		41 LBL 9		74 STO 8	
10 FRAC		42 DSZ		75 LBL 4	Ricerca delle cifre giuste al posto sbagliato
11 STO 8		43 GTO 5		76 STO 0	
12 LBL 0		44 LBL 1	Sub. per la memorizz. del tentativo di decifraz. e azzeramento R5 e R6	77 RCL i	
13 RCL 8		45 STO+i		78 STO 5	
14 RCL.0		46 CRL. X		79 RCL 9	
15 x		47 ROLL D.		80 STO 0	
16 FRAC		48 DSZ		81 LBL5	
17 STO 8		49 GTO 1		82 5	
18 LAST x		50 STO 5		83 RCL i	
19 RCL 7		51 STO 6		84 RCL.0	
20 ÷		52 4		85 $x > y ?$	Test "Flag"
21 INT		53 STO 0		86 GTO 8	
22 STO i		54 LBL 2	Ricerca delle cifre giuste al posto giusto	87 RCL 0	
23 DSZ		55 RCL i		88 $x \neq y ?$	
24 GTO 0		56 RCL.0		89 GTO 9	
25 RCL 9		57 STO+0		90 1	
26 STO 0		58 ROLL D.		91 STO-8	
27 RCL 6	Visualizz. totalizz. B By-pass per non increm. Il totalizz. N Ric. cifre gius. al pos. sbagliato	59 RCL i		92 RCL 8	
28 R/S		60 $x \neq y ?$		93 $x \neq 0 ?$	Test per usc. loop LBL 4 Azzeramento R.4 R.3 R.2 e R.1
29 LBL 6		61 GTO 6		94 GTO 4	
30 RCL.0		62 1	Incremento totalizzatore N e attiva- zione "Flag"	95 STO.4	
31 STO-0		63 STO+5		96 STO.3	
32 GTO 3		64 10^x		97 STO.2	
33 LBL 8		65 STO+i		98 STO.1	
34 ROLL D.		66 STO-0			

Figura 4. Stretto nei 98 passi, l'interessante programma che rende la HP-29C un attento Codificatore.

C.S. ma, essendo 0 il numero contenuto nel registro X, si ottiene una sequenza di cifre identica a quella generata dal calcolatore all'inizio del gioco.

Infine il programma si ferma al passo 27 dopo aver inizializzato R0 con 14 (puntatore cifre di tentativo) e visualizzato R6 (tot. B).

Quando, dopo un certo numero di tentativi, viene indovinato il C.S. si ha come risposta del calcolatore R5=4 (TOT. N=4).

A questo punto, prima di giocare un'altra partita, è necessario far proseguire l'elaboratore anche se già sappiamo che R6=0 (tot. B=0), perché quest'ultima parte del programma serve, come abbiamo visto, ad inizializzare tutto il sistema. ■

REGISTRI					
0 Usato	1 A	2 B	3 C	4 D	5 T.N./Usato
6 T.B.	7 Cost. (1.7)	8 Usato	9 Cost. (14)	10 Cost. (10)	11 a
12 b	13 c	14 d	15 Gen Codice	16 /	17 /
18 /	19 /	20 /	21 /	22 /	23 /
24 /	25 /	26 /	27 /	28 /	29 /

MERKEL SRL

MINI E MICRO COMPUTERS

Corsi di programmazione BASIC

Software per il TI 99/4A :

Pacchetti di ingegneria civile

Telai-Legge 373-Piastre-Precompressi-Ecc.

Finanziaria-Condominio-Calcolo numerico-Word P.

Ricerca Operativa-Inventario-Clienti e Fornitori.

ASSEMBLATORE PER VIC 20 E CBM 64.....£ 45.000

SISTEMA TOTOCALCIO PER CBM 64.....£ 50.000

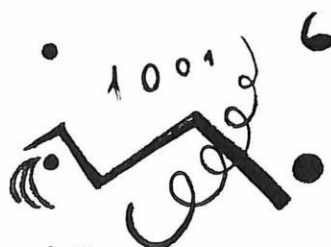
DATA BASE PER VIC 20 E CBM 64.....£100.000

Rappresentazione Grafica di Funzioni e ingrandimento
in un intervallo definito per SPECTRUM.....£ 35.000

I programmi sono disponibili su nastri e dischi.

Scrivere a MERKEL srl Via Luigia Sanfelice,7A (NA)

TEL. 081/ 241866-245278



Interfacciamento: TRS 80 pocket contro TRS 80 mod. 1

a cura della Redazione

Gli ultimi cinque anni hanno visto una curiosa rivoluzione affiorare nell'editoria specializzata in hobby tecnico-pratici. Quanti di voi ricordano le prime riviste di radiotecnica, molto spesso chiamate genericamente a coprire interessi "pratici" più che "elettronici", sentiranno certamente una certa nostalgia di ingombranti filature volanti tra valvola e valvola. La maggior parte dei circuiti riguardavano l'ambiente radioamatoriale, quindi degli amplificatori per alta e bassa frequenza, e pochi altri settori dedicati sporadicamente al controllo o alla strumentazione.

L'uso dei transistor appariva alquanto costoso e abbastanza inusuale, anche perché esisteva una certa ritrosia ad utilizzare quei piccoli involucri verniciati di nero siglati OC, che in parte o in toto potevano sostituire gli ingombranti bulbi valvolari, ma che istintivamente davano un poco di soggezione.

Ora i tredicenni appassionati di elettronica, come si autodefiniscono, non usano saldatore se non per assemblare complicatissimi, nei confronti dei circuiti ante-integrationem..., circuitini tuttofare con millepiedi sparsi un po' ovunque. Non c'è più rivistina, per quanto umile sia, che non tratti dense schede di microprocessori per controllare anche le situazioni più banali, ma che certamente offrono spunti e occasio-

ni per l'approfondimento divertente di una materia così attuale com'è lo scambio di informazioni digitali. Ogni ragazzino che si rispetti ha certamente in mano una calcolatrice, quasi sicuramente programmabile se non altro per distinguersi dagli altri colleghi scolastici.

Altrettanto vero è che se tale ragazzino abbina alla propria vanità anche il culto idolatra per tutto ciò che assomiglia ad un integrato o ad un microcomputer, certamente ha già smontato e rimontato la sua calcolatrice almeno cinque volte, la sesta rendendosi conto che non c'è nulla da fare. Con i circuiti custom la fantasia dello sperimentatore in erba si affloscia sulle insormontabili difficoltà di capire compiutamente quali segnali escono dai diversi piedini e come prelevarli senza distruggere il regalo di Natale.

In genere, peraltro, alcuni pocket computer utilizzanti il BASIC hanno una serie di uscite adattabili a differenti periferiche appositamente costruite. Così, quasi senza dubbio, è possibile collegare delle piccole stampantine, delle interfacce per registratore a cassetta e... null'altro.

È possibile sfruttare tali uscite per comandare qualcosa di differente? In linea teorica, naturalmente, la risposta è positiva.

Questo articolo descrive come, nella pratica, si possa far comunicare un pocket in BASIC di larghissi-

ma diffusione con un personal computer che, in questo caso della stessa famiglia, potrebbe essere sostituito da qualsiasi altro.

L'utilizzazione dell'interfaccia proposta è rivolta a rendere più comoda la stampa di un TRS 80 pocket (equivalente ad un PC 1211). Infatti, trasferiti i dati da stampare al TRS 80 MOD 1, si sfrutteranno le potenzialità di questo per ottenere tabulati di normali dimensioni.

Hardware e software

Come per tutti i microcomputer è assolutamente necessario prendere in esame entrambi gli aspetti di progettazione sistemistica.

La realtà hardware è relativamente semplice da implementare. In ultimo otterremo una scatoletta con pochi integrati che fungono da interfaccia tra il pocket e il mod. 1, ovvero che adeguino i segnali di uscita dell'uno alle porte di ingresso dell'altro.

Il problema software deve rispettare alcune condizioni. A parte la complicazione apparente del creare routine in linguaggio macchina che accettino i dati uscenti dal pocket, la grande limitazione cui è necessario sottostare sono le istruzioni di uscita disponibili sul pocket stesso.

Faremo uso della stessa istruzione


```

100 BREAK EQU 2 :BIT TASTO BREAK
110 CORNER EQU 3C3FH :ANGOLO SCHERMO
120 EOF EQU 0F0H :CIDICE FINE REGISTRAZIONE
130 FIRST EQU 0F100H :PRIMO BYTE BUFFER
140 NEWDOS EQU 402DH :ENTRY NEWDOS
150 PORT EQU 00H :PORTA INPUT CASSETTA
160 RETARG EQU 0A9AH :RETURN AL BASIC
170 ROW7 EQU 3840H :RIGA 7 TASTIERA
180 ORG 0F000H :61440 DECIMALE
190 PCREAD DI :DISABILITA INTERRUPT
200 LD IX,0 :ZERO IX
210 ADD IX,SP :SALVA SP IN IX
220 LD HL,FIRST-1 :PUNTA A BUFFER-1
230 CALL RDBYTE :LEGGE BYTE DI CHECKSUM
240 LD E,0 :CHECKSUM ZERO
250 CALL RDBB :LEGGE PASSWORD
260 RDBOB LD E,0 :CHECKSUM ZERO
270 LD B,10 :LEGGE 10*8 BYTE
280 NXT8 CALL RDBB :LEGGE 8 BYTE
290 DJNZ NXT8 :LEGGE IL PROSSIMO GRUPPO
300 JR RDBOB :CHECKSUM ZERO
310 RDBB PUSH BC :SALVA BC
320 LD B,8 :LEGGE 8 BYTE
330 RDBYCK CALL RDBYTE :LEGGE BYTE
340 LD (CORNER),A :DISPLAY SU SCHERMO
350 INC HL :SPOSTA PUNTATORE BUFFER
360 LD (HL),A :BYTE NEL BUFFER
370 CP EOF :E' CODICE FINE?
380 JR Z,DONE :SI SALTA
390 AND OFH :NIBBLE MENO SIGNIFICATIVO
400 LD C,A :SALVA IN C
410 AND OFH :SPOSTA BYTE
420 ADD A,E :NIBBLE PIU' SIGNIFICATIVO
430 JR NC,NOOV :SOMMA NPS A CHECKSUM
440 INC A :SKIP SE NON OVERFLOW
450 INC A :INCREMENTA CHECKSUM
460 NOOV ADD A,C :SOMMA NMS A CHECKSUM
470 LD E,A :AGGIORNA CHECKSUM
480 DJNZ RDBYCK :LEGGE PROSSIMO BYTE
490 CALL RDBYTE :LEGGE BYTE CHECKSUM
500 CP E :E' CORRETTO?
510 JR NZ,ERROR :NO SALTA
520 POP BC :RESTORE BC

530 RET :RETURN
540 ERROR LD HL,-1 :ERRORE DI CARICAMENTO
550 JR RSTSP :SALTA
560 DONE LD HL,0 :CARICAMENTO OK
570 RSTSP LD SP,IX :RESTORE SP
580 EI :ABILITA INTERRUPT
590 JP RETARG :RITORNA AL BASIC
600 RDBYTE PUSH BC :SALVA BC
610 LD C,0 :BYTE ZERO
620 CALL RDNIB :LEGGE NIBBLE PS
630 CALL RDNIB :LEGGE NIBBLE MS
640 LD D,A :SPOSTA BYTE
650 RRCA :SPOSTA FINO A CHE CORRETTO
660 RRCA :
670 RRCA :RUOTANDO
680 RRCA :QUATTRO VOLTE
690 POP BC :RESTORE BC
700 RET :RETURN
710 RDNIB LD A,(ROW7) :LEGGE PORTA
720 BIT BREAK,A :IL TASTO BREAK E' PREMUTO?
730 JR NZ,ERROR :SI SALTA
740 IN A,(PORT) :LEGGE PORTA
750 BIT 0,A :IL BIT 0 E' OFF?
760 JR NZ,RDNIB :NO
770 LD D,A :SALVA I DATI DELLA PORTA
780 LD B,4 :LEGGE 4 BIT
790 NEWBIT CALL RDBIT :LEGGE BIT
800 RR :SPOSTA CARRY IN C
810 DJNZ NEWBIT :LEGGE PROSSIMO BIT
820 CALL RDBIT :RITARDO UN BIT
830 LD A,C :SPOSTA BYTE DATI
840 RET :RETURN
850 RDBIT PUSH BC :SALVA B
860 LD B,65H :CICLIO 101 VOLTE
870 SAME IN A,(PORT) :LEGGE PORTA
880 CP D :E' LAMBIATA?
890 JR NZ,CHANGE :SI SALTA
900 DJNZ SAME :NO PROVA ANCORA
910 CHANGE LD D,A :SALVA I DATI DELLA PORTA
920 RRCA :SPOSTA BIT NEL CARRY
930 POP BC :RESTORE B
940 RET :
950 END NEWDOS :

```

```

100 POKE &H40B2,&HEF: POKE &H40B1,&HFE: CLEAR 2000
110 DEFINT A-Z: DIM D$(255),P$(255): DEFUSR=&HF000
120 CMD="PCREAD/CMD"
130 FOR I=1 TO 11
140 READ A,B
150 FOR J=A TO B
160 IF I<7 THEN READ D$(J): ELSE READ T$: D$(J)=T$+" "
170 NEXT
180 NEXT
190 D$(0)=CHR$(13)
200 D$(18)=CHR$(34)
210 A$=CHR$(27)+CHR$(75)+CHR$(5)+CHR$(255)
220 REM 23=YEN, 25=PI, 26=SQR, 57=CARET, 75=EXP
230 P$(23)=A$+CHR$(168)+CHR$(104)+CHR$(62)+CHR$(104)+CHR$(168)
240 P$(25)=A$+CHR$(130)+CHR$(252)+CHR$(128)+CHR$(252)+CHR$(130)
250 P$(26)=A$+CHR$(8)+CHR$(4)+CHR$(254)+CHR$(128)+CHR$(128)
260 P$(57)=A$+CHR$(24)+CHR$(32)+CHR$(64)+CHR$(32)+CHR$(24)
270 P$(75)=A$+CHR$(254)+CHR$(254)+CHR$(146)+CHR$(146)+CHR$(146)
280 INPUT "QUANTI CARATTERI PER LINEA":W
290 IF W<11 GOTO 280
300 PRINT "1)PC LOAD 2)DISK LOAD 3)LIST ";
310 INPUT "4)PRINT 5)DISK SAVE 6)FINE":G
320 ON G GOTO 350,420,470,470,870,950
330 GOTO 300
340 REM CARICA DAL PC
350 IF (INP(0) AND 1)=1 GOTO 380
360 PRINT "ACCENDI IL POCKET COMPUTER"
370 IF (INP(0) AND 1)=0 GOTO 370
380 PRINT "CSAVE IL PROGRAMMA DEL PC"
390 IF USR(0) THEN PRINT "ERRORE DI CARICAMENTO" ELSE
PRINT "ESEGUITO CARICAMENTO"
400 GOTO 300
410 REM CARICA DAL DISCO
420 LINEINPUT "NOME DEL FILE DA CARICARE? ":F$
430 C$="LOAD "+F$
440 CMD C$
450 GOTO 300
460 REM LIST (0 PRINT)
470 M=&HF100
480 C$="": N=0: K=0
490 FOR I=1 TO 7
500 GOSUB 820
510 IF B>0 THEN C$=D$(16*L+H)+C$
520 NEXT
530 M=M+1
540 CLS: PRINT C$
550 B=PEEK(M): M=M+1
560 IF B(&HE0) GOTO 630
570 IF B(&HF0) GOTO 300
580 IF B(&HE0) THEN C$=C$+CHR$(B-&HB0) ELSE C$=" "

590 GOSUB 820
600 IF (C$<>" " OR H>0) THEN C$=C$+CHR$(H-&H30) ELSE C$=" "
610 C$=C$+CHR$(L-&H30)+": "
620 GOTO 650
630 C$=D$(B)
640 C1$=P$(B)
650 PRINT C$: REM STRINGA DI OUTPUT
660 IF G=3 GOTO 550
670 IF C$=CHR$(13) THEN LPRINT S$: S$="": N=0: K=0: GOTO 550
680 E=LEN(C$)
690 IF C1$="" THEN C$=C1$: C1$="": E=1
700 N=N+E
710 IF N>W GOTO 750
720 S$=S$+C$
730 IF (C$="" OR N=4) THEN LPRINT S$: K=N-4: S$=""
740 GOTO 550
750 IF K=0 GOTO 790
760 LPRINT: LPRINT " ";
770 N=N-K: K=0
780 GOTO 720
790 LPRINT S$: LPRINT " ";
800 S$=C$: N=4+E: K=0
810 GOTO 730
820 B=PEEK(M)
830 H=INT(B/16): L=B-16*H
840 M=M+1
850 RETURN
860 REM SALVA NEL DISCO
870 LINEINPUT "CON CHE NOME VUOI SALVARLO SUL DISCO? ":F$
880 M=&HF108
890 IF PEEK(M)<240 THEN M=M+1: GOTO 890
900 IF M(&HF698) THEN PRINT "NON C'E' NESSUN PROGRAMMA": GOTO 300
910 C$="DUMP "+F$+" F100H,"+RIGHT$(STR$(65536+M),5)+",402DH"
920 CMD C$
930 GOTO 300
940 REM FINE
950 END

960 DATA 17,29," ",QUOTE,?,!,#,%,YEN,$,PI,SQR,"",",",":","
970 DATA 48,57,(,),>,<,+,-,*,/,[,
980 DATA 64,75,0,1,2,3,4,5,6,7,8,9,,EXP
990 DATA 81,93,A,B,C,D,E,F,G,H,I,J,K,L,M
1000 DATA 94,106,N,O,P,Q,R,S,T,U,V,W,X,Y,Z
1010 DATA 130,132,>,<,<,>
1020 DATA 144,146,TO,STEP,THEN
1030 DATA 160,173,SIN,COS,TAN,ASN,ACS,ATN,EXP,LN,LOG,INT,ABS
1040 DATA SGN,DEG,DMS
1050 DATA 176,183,RUN,NEW,MEM,LIST,CONT,DEBUG,CSAVE,CLOAD
1060 DATA 192,197,GRAD,PRINT,INPUT,RADIAN,DEGREE,CLEAR
1070 DATA 208,222,IF,FOR,LET,REM,END,NEXT,STOP,GOTO,GOSUB,CHAIN
1080 DATA PAUSE,BEEP,AREAD,USING,RETURN

```

Figura 2. La routine in linguaggio macchina e il programma in BASIC per leggere i dati dall'interfaccia e per pilotare una stampante MX 80.

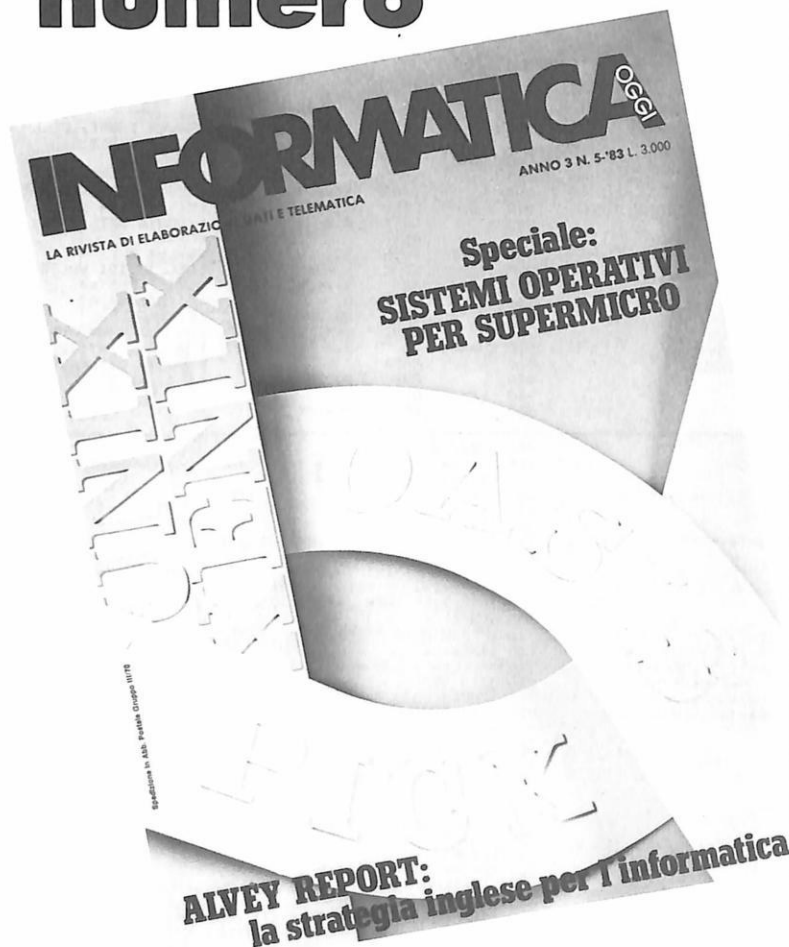
Questo è il software necessario al personal TRS 80 MOD 1 per utilizzare l'interfaccia proposta.

Il programma è autoesplicativo, e eventuali modifiche sono sempre possibili per adattarlo a personal differenti.

Il listato in linguaggio macchina, caricato in memoria, permette di accorgersi dell'inizio della trasmissione da parte del pocket, di interpretare gli impulsi uscenti e di memorizzarli sotto forma di numeri esadecimali. Il tutto rispettando il baud rate pari a 500 dei segnali provenienti dal pocket. Quando i dati sono stati letti, fa lampeggiare un cursore sul video, in alto a destra.

Il programma in BASIC preleva i numeri esadecimali memorizzati e li converte in caratteri equivalenti a quelli del pocket computer, ricostruendo le linee del BASIC del pocket.

è in edicola il nuovo numero



**SISTEMI OPERATIVI PER SUPERMICRO:
UNIX, OASIS E PICK**

**ALVEY REPORT: LA STRATEGIA INGLESE
PER L'INFORMATICA DEGLI ANNI '80**

I PROBLEMI DELL'EDP NELLE BANCHE

**CONCENTRATORI PER TERMINALI
START-STOP E BSC**

**INFORMATICA E TEORIA DEI SISTEMI:
DUE STRUMENTI AL SERVIZIO DEL MANAGEMENT**

LA CULTURA SEMIOTICA DI UN LINGUAGGIO SEMPLICE



**UNA PUBBLICAZIONE DEL
GRUPPO EDITORIALE JACKSON**

La realizzazione

Per realizzare l'interfaccia è necessario avere un poco di dimestichezza con saldatore e circuiti stampati. Un'ottima idea può essere quella di realizzarsi un circuito stampato adeguato ai componenti da inserirsi. Un altro metodo altrettanto valido è quello di montare il tutto sopra una basetta preforata, collegando con dei corti pezzetti di filo ogni reoforo dei diversi componenti, secondo lo schema dato in figura.

Per realizzare un connettore adatto al pocket computer può essere utile lavorare un poco di fantasia. Un metodo è utilizzare due fili volanti n° 22, il cui diametro è adeguato al connettore del pocket. Eventualmente, trovata la posizione ideale, sarà facile creare un piccolo blocco rigido intorno ai fili tramite l'incollaggio con un mastice a due componenti. Sarà così realizzato un artigianale, ma efficace connettore. Naturalmente chi trovasse un connettore dello stesso tipo di quelli inseriti nell'interfaccia per le cassette, ne sarebbe elegantemente avvantaggiato.

Un consiglio sempre valido è quello di dotare i due integrati di zoccolo, così che eventuali sostituzioni ne vengano facilitate e, soprattutto, non si rischi di danneggiarli durante la saldatura.

Verso il TRS 80 MOD 1 non ci sono problemi di connettori, essendo facile reperirne uno standard che si adatti alle prese del personal. L'unica avvertenza è quella di tenere i collegamenti tra interfaccia e computer i più corti possibile.

Dopo avere effettuato tutti i collegamenti necessari (per i quali ci si riferisca alle figure), data alimentazione al tutto, al comando CSAVE sul pocket il LED n° 2 dovrebbe lampeggiare. Un eventuale oscilloscopio applicato al punto di misura TP1 renderà visibile gli impulsi rettilinei uscenti dall'interfaccia.

Lanciato il programma in BASIC sul TRS 80 MOD 1, quando appare la scritta adeguata si può effettuare un CSAVE dal pocket. ■

Moduli e fasi di funzioni di trasferimento con la TI-59

di G. Fedecostante

Questo programma, generato per la calcolatrice programmabile TI-59, è stato ideato a complemento del set di programmi (orientati alla teoria dei controlli automatici) già disponibili nel package di elettronica inseribile nella suddetta calcolatrice. Accanto infatti al programma che calcola le radici reali e complesse di un polinomio (EE-09), a quello che traccia il luogo delle radici (EE-16), ed a quello che calcola la risposta nel dominio del tempo di un sistema di cui sia nota la risposta impulsiva (EE-15), un programma che calcoli anche il modulo e la fase di una funzione di trasferimento data, trova la sua naturale collocazione.

Un po' di teoria

Data una funzione di trasferimento della forma seguente

$$(1) \quad F(s) = \frac{a_0 + a_1s + a_2s^2 + a_3s^3 + \dots + a_ns^n}{b_0 + b_1s + b_2s^2 + b_3s^3 + \dots + b_ds^d}$$

calcolarne il modulo $M(\omega)$ e la fase $\varphi(\omega)$ significa innanzitutto sostituire alla variabile s la variabile immaginaria $j\omega$ ottenendo la

$$(2) \quad F(\omega) = \frac{a_0 + a_1j\omega - a_2\omega^2 - a_3j\omega^3 + \dots + a_n(j\omega)^n}{b_0 + b_1j\omega - b_2\omega^2 - b_3j\omega^3 + \dots + b_d(j\omega)^d}$$

A questo punto, raggruppando a numeratore e denominatore i termini reali ed i termini immaginari, si

perviene per $F(\omega)$ alla seguente formula finale:

$$(3) \quad F(\omega) = \frac{A_1 + j A_2}{B_1 + j B_2} \quad \text{dove} \quad \begin{aligned} A_1 &= (a_0 - a_2\omega^2 + \dots) \\ A_2 &= (a_1\omega - a_3\omega^3 + \dots) \\ B_1 &= (b_0 - b_2\omega^2 + \dots) \\ B_2 &= (b_1\omega - b_3\omega^3 + \dots) \end{aligned}$$

Essendo ora $F(\omega)$ sotto forma di rapporto di due numeri complessi è facile calcolarne il modulo e la fase tenendo presente che il modulo di

un rapporto è uguale al rapporto dei moduli mentre la fase di un rapporto è uguale alla differenza delle fasi, ovvero, nel nostro caso:

$$(4.a) \quad M(\omega) = M(2\pi f) = |F(\omega)| = \frac{\sqrt{A_1^2 + A_2^2}}{\sqrt{B_1^2 + B_2^2}}$$

$$(4.b) \quad \varphi(\omega) = \varphi(2\pi f) \quad F(\omega) = \arctg \frac{A_2}{A_1} - \arctg \frac{B_2}{B_1}$$

Il programma

Il programma (secondo il diagramma di flusso di Fig. 1), calcola, per ogni valore di frequenza impostato nel display, il modulo in dB e la fase in gradi della funzione di trasferimento (con un massimo di sei

poli e sei zeri) i cui coefficienti sono inseriti con una procedura che verrà in seguito illustrata. Per far ciò vengono utilizzate cinque subroutines definibili dall'operatore e con la seguente specifica funzione:

- A e B provvedono a memorizzare i coefficienti dei polinomi a nume-

000	14	D	059	42	STD	118	86	STF	177	43	RCL	236	22	INV
001	76	LBL	060	16	16	119	01	01	178	18	18	237	44	SUM
002	12	B	061	22	INV	120	61	GTO	179	32	X:IT	238	25	25
003	94	+/-	062	86	STF	121	01	01	180	43	RCL	239	43	RCL
004	61	GTO	063	01	01	122	33	33	181	19	19	240	23	23
005	00	00	064	92	RTN	123	43	RCL	182	22	INV	241	28	LDG
006	13	13	065	76	LBL	124	22	22	183	37	P/R	242	65	x
007	92	RTN	066	14	D	125	75	-	184	42	STD	243	02	2
008	76	LBL	067	42	STD	126	01	1	185	25	25	244	00	0
009	11	A	068	26	26	127	95	=	186	32	X:IT	245	95	=
010	87	IFF	069	02	2	128	42	STD	187	42	STD	246	22	INV
011	01	01	070	01	1	129	22	22	188	24	24	247	44	SUM
012	12	B	071	69	DP	130	22	INV	189	00	0	248	24	24
013	72	ST*	072	04	04	131	86	STF	190	32	X:IT	249	22	INV
014	17	17	073	43	RCL	132	01	01	191	43	RCL	250	86	STF
015	01	1	074	26	26	133	01	1	192	25	25	251	01	01
016	44	SUM	075	69	DP	134	44	SUM	193	77	GE	252	22	INV
017	17	17	076	06	06	135	16	16	194	02	02	253	86	STF
018	44	SUM	077	00	0	136	44	SUM	195	03	03	254	02	02
019	14	14	078	42	STD	137	17	17	196	85	+	255	00	0
020	44	SUM	079	16	16	138	43	RCL	197	03	3	256	42	STD
021	16	16	080	42	STD	139	23	23	198	06	6	257	18	18
022	43	RCL	081	17	17	140	32	X:IT	199	00	0	258	42	STD
023	16	16	082	01	1	141	43	RCL	200	95	=	259	19	19
024	32	X:IT	083	08	8	142	17	17	201	42	STD	260	42	STD
025	02	2	084	42	STD	143	22	INV	202	25	25	261	20	20
026	22	INV	085	22	22	144	77	GE	203	43	RCL	262	42	STD
027	67	EQ	086	43	RCL	145	00	00	204	24	24	263	21	21
028	00	00	087	15	15	146	00	90	205	28	LDG	264	03	3
029	39	39	088	42	STD	147	67	EQ	206	65	x	265	03	3
030	87	IFF	089	23	23	148	00	00	207	02	2	266	02	2
031	01	01	090	73	RC*	149	90	90	208	00	0	267	03	3
032	00	00	091	16	16	150	87	IFF	209	95	=	268	02	2
033	45	45	092	65	x	151	02	02	210	42	STD	269	04	4
034	86	STF	093	53	(	152	15	E	211	24	24	270	69	DP
035	01	01	094	02	2	153	00	0	212	43	RCL	271	04	04
036	00	0	095	65	x	154	42	STD	213	20	20	272	43	RCL
037	42	STD	096	89	x	155	17	17	214	32	X:IT	273	25	25
038	16	16	097	65	x	156	43	RCL	215	43	RCL	274	69	DP
039	43	RCL	098	43	RCL	157	14	14	216	21	21	275	06	06
040	14	14	099	26	26	158	75	-	217	22	INV	276	32	X:IT
041	75	-	100	54	)	159	01	1	218	37	P/R	277	03	3
042	01	1	101	45	YX	160	95	=	219	42	STD	278	00	0
043	95	=	102	43	RCL	161	42	STD	220	22	22	279	01	1
044	91	R/S	103	17	17	162	23	23	221	32	X:IT	280	06	6
045	22	INV	104	95	=.	163	02	2	222	42	STD	281	01	1
046	86	STF	105	74	SM*	164	00	0	223	23	23	282	04	4
047	01	01	106	22	22	165	42	STD	224	00	0	283	69	DP
048	61	GTO	107	87	IFF	166	22	22	225	32	X:IT	284	04	04
049	00	00	108	01	01	167	22	INV	226	43	RCL	285	43	RCL
050	36	36	109	01	01	168	86	STF	227	22	22	286	24	24
051	92	RTN	110	23	23	169	01	01	228	77	GE	287	69	DP
052	76	LBL	111	43	RCL	170	86	STF	229	02	02	288	06	06
053	13	C	112	22	22	171	02	02	230	36	36	289	91	R/S
054	42	STD	113	85	+	172	61	GTO	231	85	+	290	98	ADV
055	15	15	114	01	1	173	00	00	232	03	3	291	81	RST
056	00	0	115	95	=	174	90	90	233	06	6	292	92	RTN
057	42	STD	116	42	STD	175	76	LBL	234	00	0			
058	14	14	117	22	22	176	15	E	235	95	=			

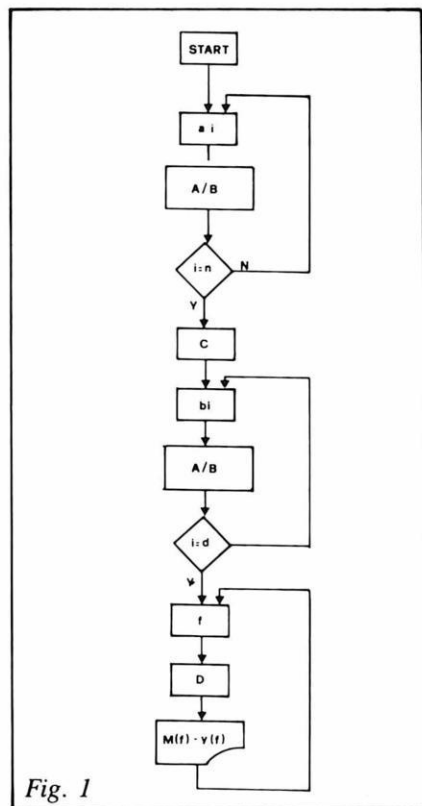


Fig. 1

ratore e denominatore con il segno corretto; come si può infatti desumere dalla (2), l'operazione di cambiamento di variabile da s a $j\omega$ cambia di segno i coefficienti dei termini di secondo e terzo grado, di sesto e settimo grado e così via, lasciando inalterati i rimanenti;

- C predispose la calcolatrice ad accettare in ingresso i coefficienti del denominatore, potendo di nuovo utilizzare A e B per la modifica del loro segno e per la loro memorizzazione;
- D calcola, per la frequenza f imposta, i valori di A_1, A_2, B_1, B_2 della (3);
- E calcola il modulo e la fase della $F(\omega)$ sulla base dei valori di A e B precedentemente trovati.

Per quanto riguarda invece le memorie, ne vengono utilizzate 27 ovvero:

- dalla 00 alla 06 per memorizzare i coefficienti del numeratore a partire dal termine noto;
- dalla 07 alla 13 per memorizzare i coefficienti del denominatore sempre a partire dal termine noto;
- la 14 e la 15 per memorizzare ri-

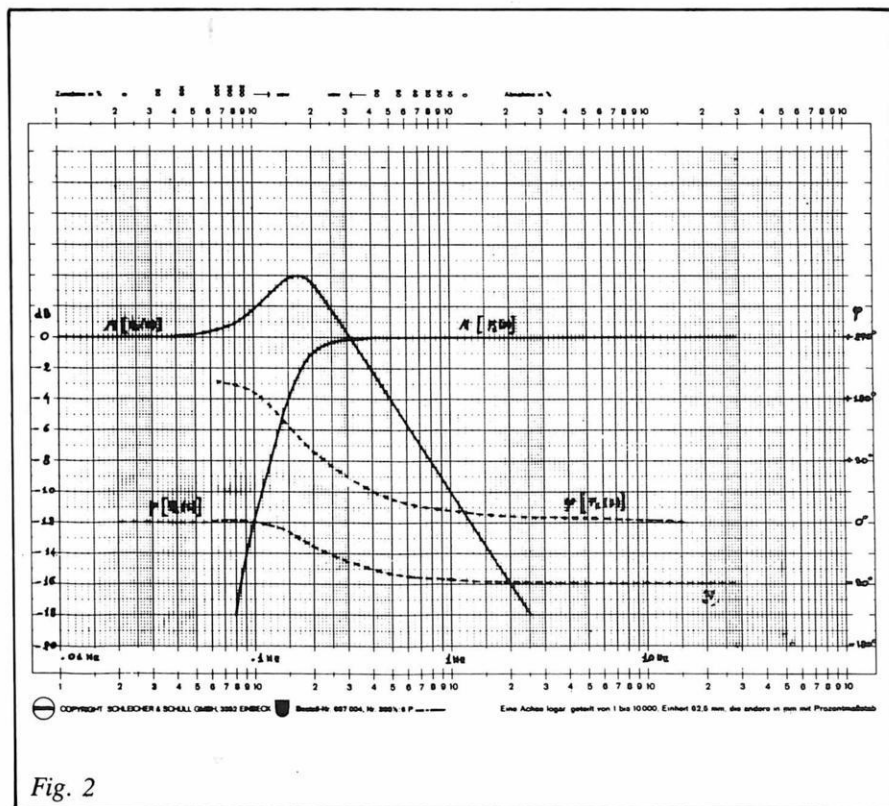


Fig. 2

spettivamente il grado del denominatore aumentato di una unità ($d+1$) e quello del numeratore (n);

- la 16 e la 17 vengono utilizzate come contatori;
- dalla 18 alla 21 per memorizzare i valori di $A_1-A_2-B_1-B_2$;
- la 22 e la 23 come registri di appoggio;
- la 24 e la 25 per memorizzare il modulo e la fase;
- la 26, infine, contiene il valore impostato per la frequenza.

Dei dieci flags disponibili sono invece usati solamente l'“1” ed il “2”. Come si può notare dal listing riportato in Fig. 2, il programma consta di 293 passi il che purtroppo, unito al fatto che usa 27 registri di memoria, non lo rende compatibile con le altre calcolatrici programmabili Texas e cioè le TI-58/58C; non è escluso però che possa in qualche modo essere ottimizzato rendendo questa compatibilità possibile.

Procedura di utilizzo

Terminata la descrizione del programma, prendiamo ora in conside-

razione la procedura per farlo girare, ed elenchiamo quindi le operazioni da effettuare:

- 1) impostare i coefficienti del polinomio a numeratore a partire dal termine noto premendo il tasto-funzione A dopo ciascuno di essi; di volta in volta apparirà sul visualizzatore il grado del termine a cui il coefficiente si riferisce;
- 2) predisporre a questo punto la calcolatrice per l'inserimento dei coefficienti del polinomio a denominatore premendo il tasto-funzione C;
- 3) ripetere il passo di cui al punto 1 per i coefficienti del denominatore;
- 4) impostare ora sul visualizzatore il valore di frequenza f desiderato;
- 5) premere il tasto-funzione D per calcolare il modulo e la fase; $M(f)$ apparirà sul display, la fase sarà contenuta nel registro t ;
- 6) per altri valori di frequenza f è sufficiente ora inserire il nuovo valore nel visualizzatore e quindi premere R/S dopodiché, come in precedenza, il modulo apparirà sul display mentre la fase sarà nel registro t .

Come esempio, in Fig. 3 sono ri-

0.05	F	0.15	F
233.3428512	PHI	143.457344	PHI
-30.17518089	MDB	-3.850399449	MDB
0.06	F	0.16	F
225.621096	PHI	134.2414814	PHI
-25.43258228	MDB	-2.941852463	MDB
0.07	F	0.17	F
217.6549245	PHI	125.5969273	PHI
-21.43465003	MDB	-2.235813143	MDB
0.08	F	0.18	F
209.3888664	PHI	117.6204945	PHI
-17.993299	MDB	-1.696284705	MDB
0.09	F	0.19	F
200.7748853	PHI	110.3482452	PHI
-14.99439403	MDB	-1.288711543	MDB
0.1	F	0.5	F
191.7851325	PHI	37.16081081	PHI
-12.36852616	MDB	-.0045150161	MDB
0.11	F	1.	F
182.4285014	PHI	18.31710551	PHI
-10.07497092	MDB	-.0000705802	MDB
0.12	F	2.	F
172.767248	PHI	9.128603692	PHI
-8.090838899	MDB	-.0000011029	MDB
0.13	F	5.	F
162.9262359	PHI	3.648179319	PHI
-6.401967306	MDB	-.0000000046	MDB
0.14	F	10.	F
153.0861622	PHI	1.823858325	PHI
-4.994812385	MDB	-.0000000001	MDB

portati i diagrammi del modulo e della fase, ottenuti con questo programma, relativi alle due seguenti funzioni:

$$F_1(s) = \frac{s^3}{s^3 + 2s^2 + 2s + 1}$$

$$F_2(s) = \frac{2s^2 + 2s + 1}{s^3 + 2s^2 + 2s + 1}$$

Da notare, a conclusione, come questo programma preveda il possibile uso della stampante PC 100 per cui è possibile ottenere anche un print automatico dei valori di modulo e fase calcolati e della frequenza impostata come riportato nell'esempio di Fig. 4.

Immaginate di fornire al vostro computer le seguenti affermazioni: GLI UOMINI SONO ANIMALI, GLI UOMINI SONO RAZIONALI e che il computer deduca prontamente che evidentemente QUALCHE ANIMALE È RAZIONALE. Fantascienza? C'è forse da temere che il computer "impazzito" cominci a farneticare frasi del tipo COGITO ERGO SUM? Niente di tutto questo, naturalmente.

In questo articolo spiegheremo come "insegnare" all'elaboratore a compiere alcuni ragionamenti di tipo logico. Per insegnare un po' di logica al computer non c'è bisogno che questi si chiami Hal (il famosissimo mega-elaboratore di "2001 Odissea nello spazio"), ma basta un minuscolo pocket come il Casio FX-702P. Non c'è nemmeno bisogno di conoscere sofisticatissimi linguaggi di programmazione: è sufficiente l'elementare Basic. Infine non è necessario essere degli studiosi di logica matematica o di intelligenza artificiale, ma basteranno le poche nozioni di logica aristotelica e di algebra booleana che ora daremo.

Cos'è un sillogismo

Il tipo di ragionamento usato all'inizio dell'articolo prende il nome di *sillogismo* ed il nostro scopo è appunto quello di programmare il calcolatore in modo che sia in grado di risolvere qualsiasi tipo di sillogismo. Un sillogismo consiste di tre proposizioni: le prime due (dette *premesse*) devono essere tali che la loro verità implichi la verità della terza (detta *conclusione*). Noi forniremo all'elaboratore le premesse ed esso dovrà estrapolarne la conclusione o, se le premesse sono errate, perché non permettono di formulare una conclusione logicamente corretta, dovrà segnalare l'errore.

Consideriamo, per esempio, i seguenti tre sillogismi:

1) TUTTI GLI UOMINI SONO ANIMALI (1a premessa)

Logica aristotelica con la calcolatrice tascabile

di Giuseppe De Nicolao

(soggetto) (medio)
TUTTI GLI ANIMALI SONO
MORTALI (2a premessa) dunque
(medio) (predicato)
TUTTI GLI UOMINI SONO
MORTALI (conclusione)
(soggetto) (predicato)
2) OGNI UOMO È ANIMALE (1a
premissa)
(soggetto) (medio)
NESSUNA PIANTA È ANIMALE
(2a premessa) dunque
(predicato) (medio)
NESSUN UOMO È PIANTA (con-
clusione)
(soggetto) (predicato)
3) QUALCHE UOMO È FILOSO-
FO (1a premessa)
(soggetto) (medio)
TUTTI I FILOSOFI SONO SA-
PIENTI (2a premessa) dunque
(medio) (predicato)
QUALCHE UOMO È SAPIENTE
(conclusione)
(soggetto) (predicato)

Notiamo innanzitutto che le proposizioni che formano i sillogismi si caratterizzano per il fatto di associare o dissociare (per esempio il soggetto "pianta" viene dissociato dal predicato "animale") un *soggetto* ed un *predicato*. Caratteristica essenziale del sillogismo è l'esistenza di un termine comune alle due premesse (detto termine *medio* del sillogismo) che ha la funzione di permettere la connessione logica tra i termini (soggetto e predicato) della conclusione, che devono apparire ciascuno una sola volta nelle premesse.

Le singole proposizioni del sillogi-

smo vengono poi classificate in *affermative* o *negative* a seconda che congiungano o disgiungano un soggetto ed un predicato (per esempio: QUALCHE UOMO È PAZIENTE è affermativa, NESSUN UOMO È PIANTA è negativa). Un'altra classificazione è quella che divide le proposizioni in *universali* (TUTTI GLI UOMINI SONO MORTALI oppure, il che è lo stesso, OGNI UOMO È MORTALE; NESSUN UOMO È PIANTA oppure, il che è lo stesso, OGNI UOMO È NON PIANTA) e *particolari* (QUALCHE UOMO È SAPIENTE). Le proposizioni universali riguardano la totalità, quelle particolari solo una parte di un certo insieme (per esempio l'insieme degli uomini). Le proposizioni universali e particolari possono entrambe essere affermative o negative.

Se si combinano tutti questi tipi di proposizioni tra di loro (rispettando la regola di avere un termine medio comune alle due premesse e che il soggetto ed il predicato della conclusione appaiano ciascuno una sola volta) si ottengono 256 modi possibili di sillogismo di cui però solo circa una ventina sono logicamente corretti. È infatti evidente che, per esempio, il seguente sillogismo è scorretto: OGNI ETIOPE È NERO, OGNI ETIOPE È UOMO dunque OGNI UOMO È NERO. L'opera compiuta da Aristotele, che per primo studiò scientificamente il sillogismo, consistette appunto nell'individuare le forme di sillogismo valide e nel formulare le regole che

PROPOSIZIONI	affermative	negative
Universali	Ogni X è Y	Nessun X è Y
Particolari	Qualche X è Y	Qualche X è non Y

Figura 1. Classificazione delle proposizioni che formano i sillogismi.

ne garantiscono la correttezza. Noi però procederemo per un'altra strada, dando prima un'interpretazione insiemistica del ragionamento sillogistico e formalizzandolo poi in termini matematici attraverso l'algebra booleana. Non spaventatevi! Nulla è semplice, a questo livello.

L'algoritmo di risoluzione

Prima di passare alla rappresentazione insiemistica delle proposizioni che formano il sillogismo, sarà utile ricordare alcune semplici nozioni di teoria degli insiemi. Dati due insiemi A e B, chiameremo intersezione di A e B, e l'indicheremo con $A * B$, l'insieme i cui elementi appartengono simultaneamente ad A e a B; chiameremo insieme complementare di A e lo indicheremo con $1-A$, l'insieme formato da tutti gli elementi che non appartengono ad A (se, per esempio, M è l'insieme degli esseri mortali, $1-M$ sarà l'insieme dei non mortali, cioè l'insieme degli esseri immortali).

Immaginiamo ora che i termini delle preposizioni del sillogismo siano degli insiemi (ad esempio GLI

UOMINI starà per l'insieme degli uomini). Dire che OGNI UOMO È ANIMALE vorrà dire che l'insieme degli uomini è interamente compreso in quello degli animali, ovvero che l'intersezione tra l'insieme degli uomini (simbolo U) ed il complementare dell'insieme degli animali (simbolo $1-A$) è l'insieme vuoto: $U * (1-A) = 0$ (non esiste alcun uomo che sia non animale, perché, se esiste, sarebbe $U * (1-A) \neq 0$).

Dire che NESSUNA PIANTA È ANIMALE vorrà dire che l'intersezione dell'insieme delle piante (simbolo P) e dell'insieme degli animali (simbolo A) è l'insieme vuoto: $P * A = 0$ (non esiste niente che sia simultaneamente pianta ed animale).

Dopo aver rappresentato le proposizioni universali affermative e negative, passiamo ora alle proposizioni particolari.

Dire che QUALCHE UOMO È FILOSOFO significa che l'intersezione tra l'insieme degli uomini (simbolo U) e l'insieme dei filosofi (simbolo F) è un insieme non vuoto (che rappresenteremo con V): $U * F = V$ (esiste un insieme V non vuoto in cui elementi sono contemporaneamente uomini e filosofi). Infine, se QUALCHE UOMO NON È SAGGIO (oppure QUALCHE UOMO È NON SAGGIO), vuol dire che l'intersezione tra l'insieme degli uomini e l'insieme dei non saggi (che è l'insieme complementare dei saggi ed ha quindi simbolo $1-S$) è un insieme non vuoto (che rappresenteremo con W): $U * (1-S) = W$.

A partire da questa rappresentazione insiemistica delle proposizioni è possibile ottenere una rappresentazione grafica del ragionamento sillogistico, la quale permette di verificare facilmente la correttezza o la scorrettezza del ragionamento (vedi figura 3).

Se ora noi diamo alle lettere che simboleggiavano gli insiemi il significato di variabili logiche o booleane (ovvero di variabili che possano assumere solo i valori FALSO e VERO rappresentati rispettivamente da 0 ad 1) la rappresentazione insiemistica diventa una formalizzazione matematica.

La variabile logica U assumerà valore VERO (sarà cioè $U=1$) o

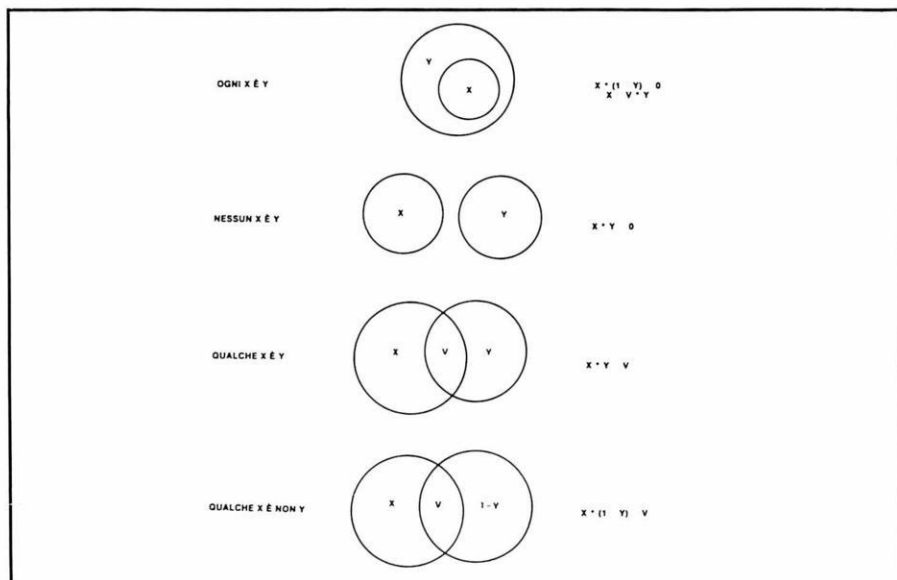
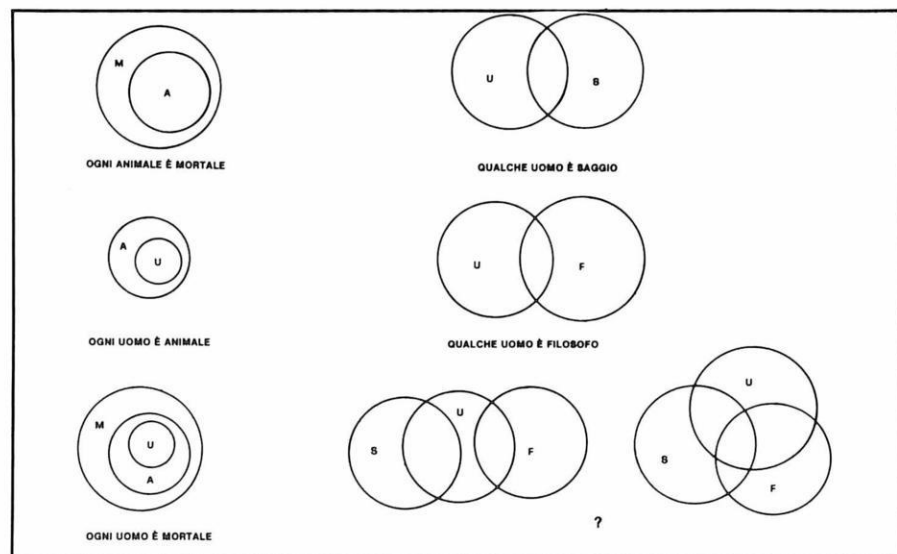


Figura 2. Rappresentazione insiemistica e logica delle proposizioni universali affermative e negative e di quelle particolari affermative e negative.

Figura 3. Rappresentazione grafica del ragionamento sillogistico. Il sillogismo a sinistra è corretto mentre quello a destra è scorretto perché non si può ricavare una relazione necessaria tra i saggi e i filosofi.



FALSO (sarà cioè $U=0$) a seconda che qualcosa appartenga o no all'insieme UOMINI. In questo modo le proposizioni si trasformano in equazioni logiche ed il sillogismo si trasforma in un sistema di due equazioni logiche in tre variabili (medio, soggetto e predicato). Consideriamo ad esempio il sillogismo già visto.

OGNI UOMO È ANIMALE
 $U * (1-A) = 0$
OGNI ANIMALE È MORTALE

$$A * (1-M) = 0$$

OGNI UOMO È MORTALE

Ciascuna delle premesse si trasforma in una funzione logica $W(U, A)$ e $W(A, M)$ in due variabili. Ponendo $U=1$ potremo ricavare procedendo per tentativi (A infatti può assumere solo i valori 0 ed 1) il valore di A che soddisfa l'equazione e ricavare allo stesso modo per tentativi il valore di M.

Se, ponendo $U=1$, si otterrà

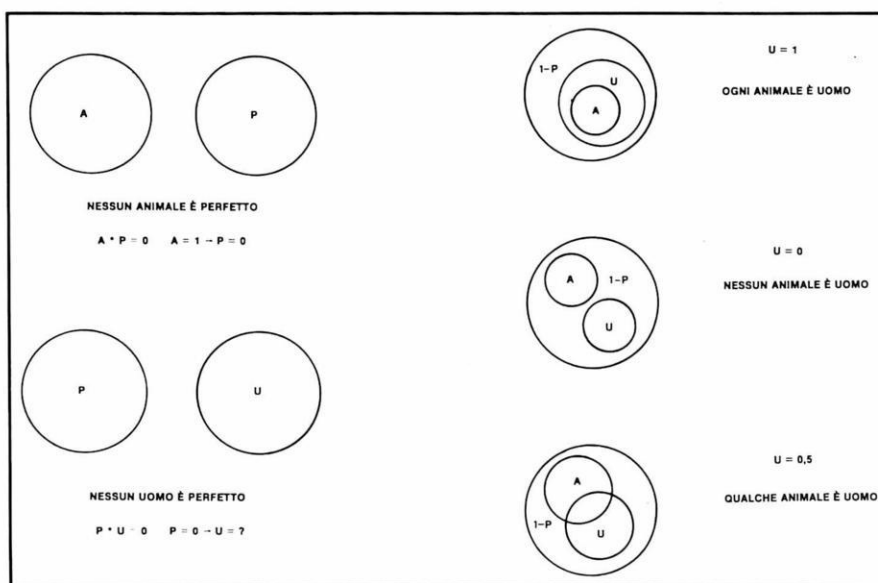


Figura 4. L'equazione logica $P*U=0$ per $P=0$ perde significato, come è mostrato in figura. Con le premesse date è possibile uno qualsiasi dei risultati $U=1$, $U=0$, $U=0,5$, perché in tutti e tre i casi l'equazione $P*U$ è verificata. Il sillogismo va quindi considerato scorretto.

$M=1$ (come in questo caso) vorrà dire che dall'appartenere all'insieme degli uomini segue necessariamente l'appartenenza all'insieme dei mortali (segue cioè necessariamente che la variabile M assume valore 1, che sta per VERO): in altre parole OGNI UOMO È MORTALE (se fosse invece risultato $M=0$, ciò avrebbe significato che NESSUN UOMO È MORTALE).

Boole per risolvere sistemi di equazioni logiche, ed in particolare per risolvere sillogismi, usava procedimenti diversi, i quali però risultano molto più difficilmente applicabili ad un elaboratore.

Sembrerebbe a questo punto di aver trovato l'algoritmo giusto per risolvere qualsiasi sillogismo, ma purtroppo le cose non stanno così. Per rendercene conto consideriamo il seguente sillogismo:

QUALCHE UOMO È FILOSOFO
 $U*F=V$

TUTTI I FILOSOFI SONO
SAPIENTI $F*(1-S)=0$
QUALCHE UOMO È SAPIENTE

Se poniamo $U=1$ ci troviamo nell'impossibilità di ricavare il valore di F della prima equazione a causa dell'insieme V , di cui non sappiamo niente se non che è un insieme non vuoto. D'altra parte, se consideriamo la rappresentazione insiemistica delle preposizioni particolari affermative, ci rendiamo conto che l'appartenenza all'insieme dei filosofi non è totale, e neppure quella all'insieme dei non-filosofi. Esprimiamo matematicamente questo fatto attribuendo alla variabile V il valore 0,5: in questo modo dalla verità di U (cioè dall'essere uomini) segue la "semi-verità" di F (infatti il valore di F che soddisfa l'equazione è 0,5).

Tenendo presente la "semi-verità" di F , poniamo $F=1$ nella seconda premessa e ricaviamo (sempre per tentativi) che $S=1$.

COME USARE IL PROGRAMMA

Prima di utilizzare il programma si deve eseguire l'operazione DEFM 1 (infatti nel programma compaiono le variabili A(5) e A(6). Le frasi che verranno inserite nel programma dovranno rispettare il seguente schema:

OGNI

QUALCHE "SOGGETTO" È (NON) "PREDICATO"

NESSUNO (0/A)

Consideriamo il seguente esempio (si suppone che il programma sia memorizzato in P0):

TASTI PREMATURI		DISPLAY	
PO		FRASE 1	?
OGNI	EXE		?
UOMO	EXE		?
E	EXE		?
ANIMALE	EXE	FRASE 2	?
NESSUN	EXE		?
ANIMALE	EXE		?
E	EXE		?
PERFETT	EXE	NESSUNO UOMO È PERFETT	

(si noti che si è scritto PERFETT e non PERFETTO, perché la Casio non accetta stringhe lunghe più di 7 caratteri).

Consideriamo ora un sillogismo scorretto:

TASTI PREMATURI		DISPLAY	
PO		FRASE 1	?
QUALCHE	EXE		?
UOMO	EXE		?
E	EXE		?
SAGGIO	EXE	FRASE 2	?
QUALCHE	EXE		?
UOMO	EXE		?
E	EXE		?
FILOSOF	EXE	SCORRETTO	

Se la prima premessa è del tipo QUALCHE UOMO È NON PERFETTO, si deve considerare NON PERFETTO come un predicato unico e trasformare la premessa in QUALCHE UOMO È IMPERFETTO (il programma non accetta come prime premesse proposizioni particolari negative, che quindi dovranno sempre essere trasformate come si è appena visto).

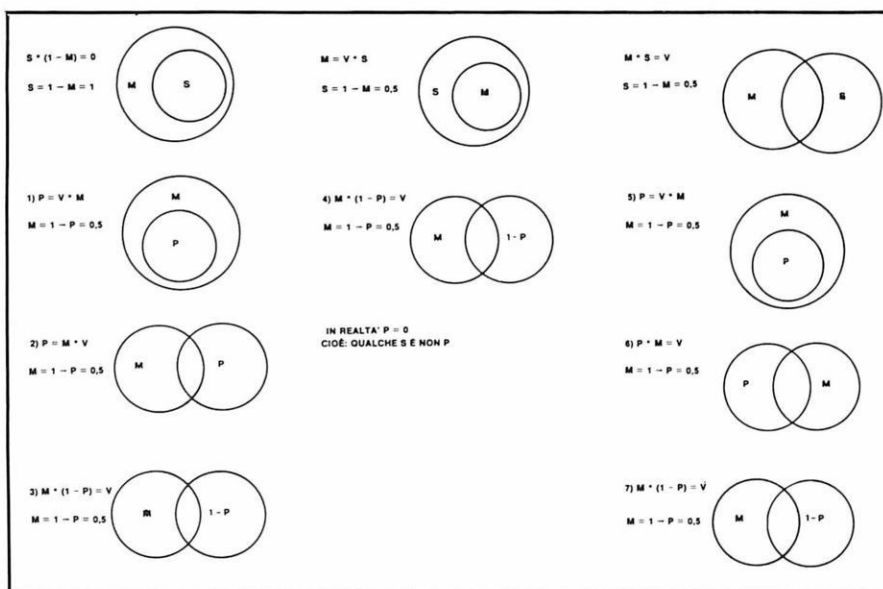
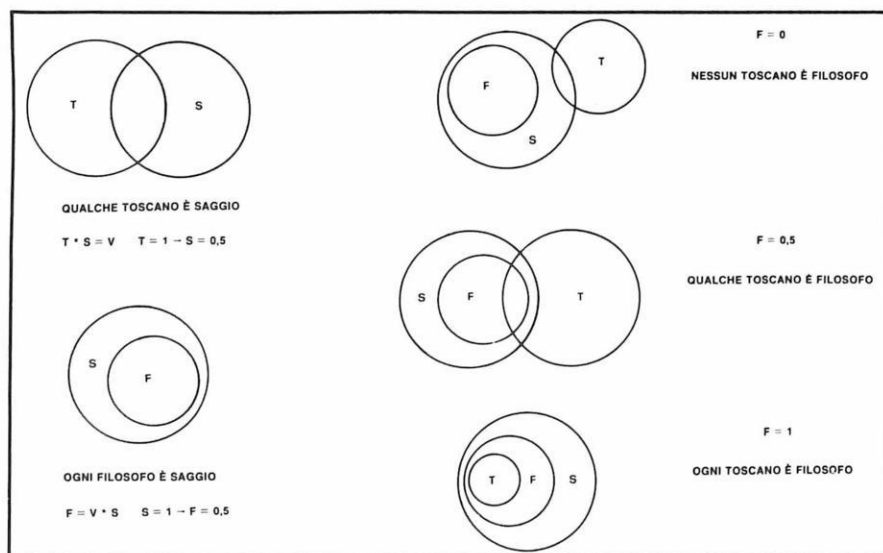


Figura 5. Elenco dei 7 casi in cui il procedimento di risoluzione darebbe luogo a risultati errati. Nel caso 4 risulta $P=0,5$ invece che $P=0$ mentre in tutti gli altri casi, il sillogismo è scorretto. S sta per soggetto, M per medio, P per predicato.

Figura 6. Rappresentazione grafica di un caso in cui il procedimento dà un risultato errato: si ottiene $F=0,5$, ma in realtà il sillogismo è scorretto, poiché sono possibili le tre soluzioni $F=0$, $F=1$, $F=0,5$ senza che le premesse siano contraddette.



Cosa significherebbe $F=0,5$ e $S=1$? Significherebbe che quella parte di uomini che sono filosofi sono necessariamente sapienti, cioè QUALCHE UOMO È SAPIENTE.

Nell'applicazione di questa metodologia sorgono però delle altre difficoltà, consideriamo per esempio il sillogismo:

OGNI UOMO È ANIMALE
 $U \cdot (1-A) = 0$

QUALCHE UOMO È SAGGIO
 $U \cdot S = V$
QUALCHE ANIMALE È SAGGIO

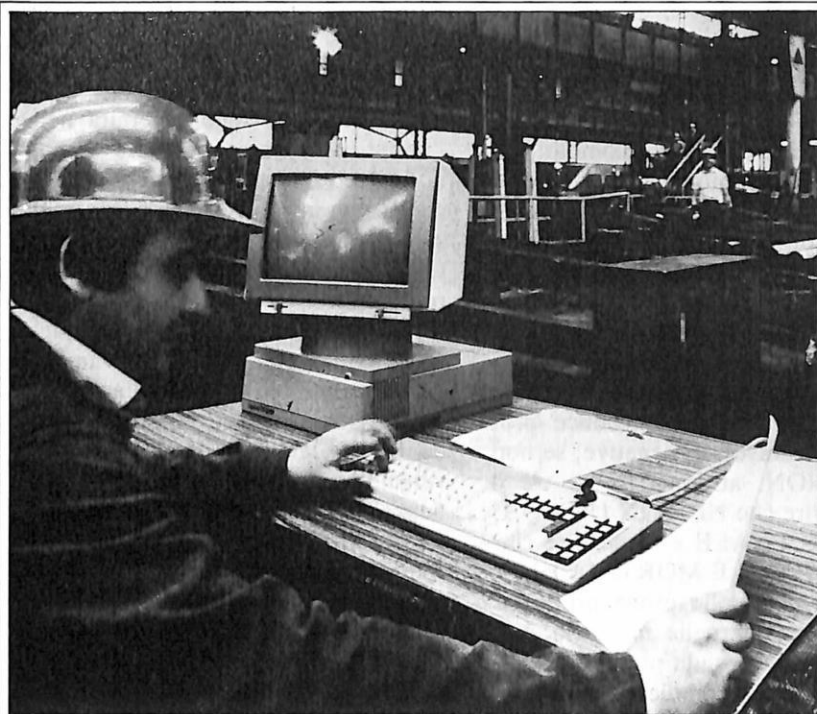
Il termine medio è U; se poniamo, come negli altri casi, il soggetto (cioè A) = 1 la prima equazione si riduce ad un'identità e perde significato. Generalizzando il problema ciò vuole dire che la funzione logica $X \cdot (1-Y)$ va bene quando la variabile

indipendente è X e si tratta di ricavare il valore di Y che soddisfa l'equazione (ovvero, ricordando la rappresentazione insiemistica delle proposizioni universali affermative, quando la variabile indipendente è l'insieme incluso e la variabile dipendente è l'insieme includente).

Nel caso in cui la variabile indipendente sia l'appartenenza o la non appartenenza all'insieme includente useremo l'espressione $Y = V \cdot X$ (dove V ha il significato logico di "parte di" ed ha valore numerico 0,5). Consideriamo di nuovo il sillogismo in questione: se la prima equazione diventa $U = V \cdot A$, l'appartenere agli insiemi degli animali ($A=1$) implica la semi verità di U (cioè tutte le proprietà che deriveranno necessariamente dall'appartenere all'insieme degli uomini saranno anche proprietà non di tutti gli animali, ma di alcuni di essi). Trovata la "semi-verità" di U, sostituiranno $U=1$ nella seconda premessa ottenendo la "semi-verità" di S: ciò vorrà dire che una parte degli uomini, che sono una parte degli animali, gode delle proprietà di essere saggio, cioè che QUALCHE ANIMALE È SAGGIO.

In alcuni casi le equazioni delle premesse per particolari valori delle variabili indipendenti potranno trasformarsi in identità (per esempio $X \cdot Y = 0$ qualora sia $X=0$ diviene un'identità): nel caso che una delle premesse risulti verificata per più di un solo valore della variabile dipendente, il programma segnalerà la scorrettezza del sillogismo. L'equazione $X \cdot Y = 0$ sta a significare che i due insiemi X e Y sono disgiunti (ad esempio: nessun uomo è perfetto): è chiaro che, se consideriamo come dato di partenza la non appartenenza ad uno dei due insiemi (se consideriamo per esempio l'insieme degli animali come esseri non perfetti), non possiamo trarre nessuna conclusione mediante l'equazione (dal momento che gli uomini non esauriscono gli esseri imperfetti, gli animali potrebbero essere indifferentemente uomini o no e nemmeno possiamo dedurre che qualche animale è uomo; vedi la figura 4).

Applicando il metodo esposto fino a qui si può però ancora incorre-



NOVITA'!

CANARD

CORSO DI TECNICA DIGITALE

IL PROGRESSO DELL'ELETTRONICA PER IL TUO PROGRESSO PROFESSIONALE

Il minuscolo computer che regola una lavabiancheria, il video-terminale che permette di sorvegliare e di dirigere il montaggio robotizzato di un'automobile. Ecco solo due esempi dei progressi dell'elettronica. Progressi continui che richiedono la presenza di esperti in tecniche digitali nell'industria, nei servizi, nelle telecomunicazioni. Sarà proprio questo nuovo corso per corrispondenza Scuola Radio Elettra la base di partenza per inserirti in uno di questi settori o per migliorare il tuo attuale livello professionale. O, ancora, per entrare nell'affascinante mondo degli hobbisti della microelettronica. Con il metodo Scuola Radio Elettra, basato sulle esercitazioni pratiche, ti accorgerai di come studiare possa essere appassionante. Con le lezioni e i materiali che ti saranno forniti dalla Scuola e che resteranno di tua proprietà, realizzerai tutte le esperienze previste dal programma di studio e inoltre costruirai il DIGILAB, il tuo laboratorio digitale da tavolo per tanti diversi circuiti applicativi (termometro digitale, contasecondi elettronico, chiave elettronica...). Al termine del Corso un Attestato testimonierà la tua preparazione. Spedisci il tagliando. Riceverai, gratis e senza impegno, una completa documentazione a colori.

Il DIGILAB, il laboratorio digitale che rimarrà di tua proprietà.



Scuola Radio Elettra
Via Stellone 5/27C • 10126 Torino

Da trent'anni insegna il lavoro.

PER CORTESIA, SCRIVERE IN STAMPATELLO

SCUOLA RADIO ELETTRA Via Stellone 5/27C 10126 TORINO
 Contrassegna con una crocetta la casella relativa al corso o ai corsi che vi interessano.

☐ Elettronica radio TV (novità)	☐ Disegnatore meccanico progettista
☐ Radio stereo	☐ Esperto commerciale
☐ Televisione bianco e nero	☐ Impiegata d'azienda
☐ Televisione a colori	☐ Tecnico d'officina
☐ Elettrotecnica	☐ Motorista autoriparatore
☐ Elettronica industriale	☐ Assistente e disegnatore edile
☐ Amplificazione stereo	☐ Lingue
☐ Alta fedeltà (novità)	☐ Sperimentatore elettronico
☐ Fotografia	☐ Dattilografia (novità)
☐ Elettrotecnica	☐ Disegno e pittura (novità)
☐ Programmazione su elaboratori elettronici	☐ Cosmesi (novità)
	☐ Tecnica digitale (novità)

Nome _____
 Cognome _____
 Professione _____ Età _____
 Via _____ N. _____
 Località _____
 Cod. Post. _____ Prov. _____
 Motivo della richiesta: per hobby ☐ per professione o avventura ☐

Tagliando da compilare, ritagliare e spedire in busta chiusa (o incollato su cartolina postale)

re in errori (vedi la figura 5); per esempio:

QUALCHE TOSCANO È
SAGGIO T*S=V
OGNI FILOSOFO È SAGGIO
F=V*S

Applicando il metodo a questo sillogismo scorretto (vedi figura 6) si troverebbe che $S=0,5$ ed $F=0,5$, cioè che QUALCHE TOSCANO È FILOSOFO (il che, con buona pace degli amici toscani, non è affatto detto perché all'interno dell'insieme dei saggi l'insieme dei filosofi e dei toscani potrebbero essere disgiunti). Da qui sorge la necessità di disporre di una subroutine di correzione che segnali la scorrettezza dei sillogismi nei casi elencati nella figura 5.

Il programma

A questo punto, supponendo di aver scritto un programma che risolve un qualsiasi sistema di 2 equazioni logiche con il procedimento appena visto, basterebbe riservare due linee dello stesso alla scrittura delle equazioni che rappresentano il sillogismo da risolvere, ed interpretare i valori logici (1 cioè verità, 0,5 cioè "semi-verità", 0 cioè falsità) che vengono assunti dal termine medio e dal predicato. Tuttavia un programma di questo tipo richiede che l'utilizzatore sappia, per esempio, che l'equazione corrispondente a NESSUN UOMO È PERFETTO è $U*P=0$ e così via. Il programma risulta molto più comodo se l'input consiste semplicemente nell'immissione parola per parola delle due premesse del sillogismo e se i risultati non sono dei numeri, ma direttamente la conclusione del sillogismo espressa a parole. Il programma può pertanto essere scisso in tre parti: 1) immissione delle premesse parola per parola e individuazione del tipo di equazioni corrispondenti alle frasi inserite; 2) risoluzione del sistema di equazioni secondo il metodo già esposto; 3) traduzione dei risultati numerici sotto forma di una proposizione conclusiva.

Noi supponiamo che una premessa qualsiasi debba essere espressa secondo il seguente schema:

OGNI
"QUALCHE SOGGETTO" È

(NON) "PREDICATO"

NESSUNO

(non è detto che debba comparire il NON, che appunto per questo è messo tra parentesi). OGNI introduce le proposizioni universali negative ed affermative (a seconda che sia presente o no il NON). QUALCHE introduce proposizioni particolari negative o affermative (a seconda che sia presente o no il NON). NESSUNO introduce proposizioni universali negative, se non c'è il NON, affermative, se c'è il NON (dire che NESSUN UOMO È NON MORTALE è come dire che OGNI UOMO È MORTALE).

Il soggetto della prima premessa sarà contenuto nella memoria A\$, quello della seconda premessa nella memoria C\$, il predicato della prima premessa in B\$, quello della seconda in D\$ (a questo proposito è utile ricordare che la Casio FX-702P non accetta stringhe più lunghe di 7 lettere: pertanto, se si deve inserire la parola FILOSOFO, bisognerà troncarla a FILOSO, troncamento

che in ogni caso non pregiudica la chiarezza).

La parola È viene accettata nell'imput solo per rendere più "umano" l'uso del programma (è chiaro che È non aggiunge alcuna informazione). Le parole OGNI, QUALCHE, NESSUN costituiscono le parole chiave: le memorie A(5) ed A(6), che indicano il tipo di equazione corrispondente rispettivamente alla prima premessa e alla seconda, assumono valore 1 in corrispondenza di OGNI (proposizione universale affermativa), 2 in corrispondenza di QUALCHE (proposizione particolare affermativa), -1 in corrispondenza di NESSUN (proposizione universale negativa).

Se nel seguito della frase compare il NON, il valore della variabile A(5) e A(6) viene cambiato di segno (il segno negativo sta infatti per proposizioni negative, quello positivo per proposizioni affermative). Dal momento che se la prima parola non è OGNI né QUALCHE essa viene supposta automaticamente uguale a NESSUN, nel caso di sog-

LIST ALL	36 IF A\$=C\$;P\$=D\$:	96 PRT S\$;
	GSB 300:60TO 40	98 IF Z=0:PRT " NO
*** PRG LIST	38 P\$=C\$:GSB 300:6	N";
	SB 400	100 PRT " E ";P\$
VAR: 36 PRG: 1600	40 A(5)=A(5)+200:A	102 END
	(6)=A(6)+200	104 PRT "SCORRETTO"
P0: 832 STEPS	50 X=1	:END
3 FOR I=1 TO 2	52 FOR I=1 TO 2	198 W=X*(1-Y)-.5:RE
5 WAIT 20:PRT "FR	54 FOR J=0 TO 2	T
ASE ";I	56 Y=J*.5	199 W=Y*X:RET
7 INP C\$	58 GSB A(I+4)	200 W=Y-.5*X:RET
9 IF C\$="QUALCHE"	60 IF W=0:A(J)=1:A	201 W=X*(1-Y):RET
:A(I+4)=2:GOTO	(I+2)=Y:GOTO 64	202 W=X*Y-.5:RET
15	62 A(J)=0	300 IF A(5)=1:A(5)=
11 IF C\$="OGNI":A(	64 NEXT J	0
I+4)=1:GOTO 15	66 A=A(0)+A(1)+A(2	301 RET
13 A(I+4)=-1	)	400 IF A(6)=1:A(6)=
15 INP C\$	68 IF A#1 THEN 104	0
16 INP D\$	70 X=A(I+2)	401 RET
17 INP D\$	72 IF X#0:X=1	498 GOTO 80
19 IF D\$="NON" THE	74 NEXT I	500 GOTO 550
N 24	76 N=A(3):Q=A(4)	502 IF Q=.5 THEN 10
21 A(I+4)=-A(I+4)	78 GOTO ((50+A(5))	4
23 INP D\$	*2)	503 GOTO 80
24 IF I=1:A\$=C\$:B\$	80 Z=1	504 IF Q=1 THEN 80
=D\$	86 IF N=.5:PRT "QU	505 IF Q=0 THEN 80
25 NEXT I	ALCHE ";:GOTO 9	506 GOTO 104
30 IF B\$=C\$:S\$=A\$:	4	550 IF A(6)#198 THE
P\$=D\$:GOTO 40	90 IF Q=0:PRT "NES	N 80
32 IF B\$=D\$:S\$=A\$:	SUN ";:GOTO 96	551 IF P\$=D\$:Q=0:G0
P\$=C\$:GSB 400:6	92 PRT "OGNI ";:G0	TO 80
OTO 40	TO 96	552 GOTO 104
34 S\$=B\$	94 IF Q=0:Z=0	

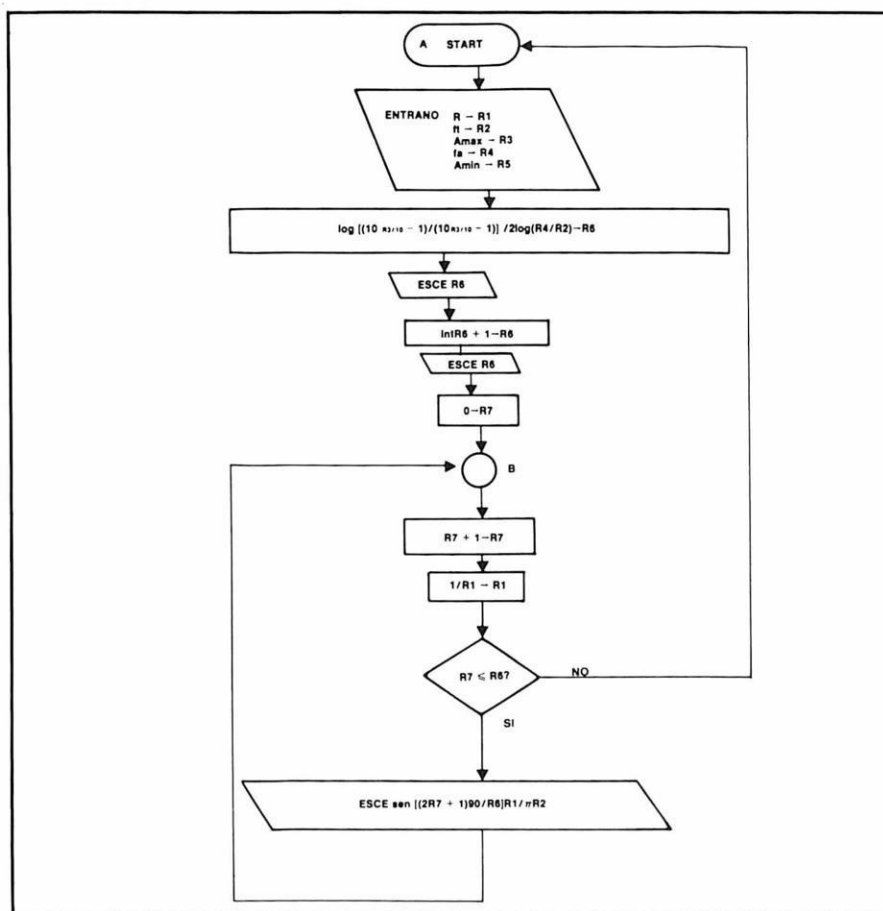


Figura 7. Diagramma di flusso del programma.

getto femminile l'uso di NESSUNA non porta ad errore.

La parte immediatamente successiva del programma individua confrontando A\$, B\$, C\$, D\$ e il termine medio, cioè quello che compare due volte nelle premesse.

Trovatolo, si suppone che il soggetto della conclusione sia il termine diverso del medio che compare nella prima premessa e che il predicato della conclusione sia il termine diverso del medio che compare nella seconda premessa. La memoria S\$ conterrà il soggetto della conclusione, la memoria P\$ il predicato della conclusione. Nel caso che nelle equazioni delle premesse di tipo universale affermativo il termine noto non sia l'insieme incluso, ma quello includente, le subroutine 300 e 400 provvedono a porre A(5) e A(6) = 0 (dove 0 sta per proposizione universale affermativa in cui il termine noto è l'insieme includente).

Nella parte successiva del pro-

gramma, dedicata alla risoluzione del sistema, le variabili A(5) ed A(6) diventano l'indirizzo delle subroutine che contengono tutti i diversi tipi di equazioni possibili (ogni equazione è scritta nella forma $W = f(XN Y)$, dove X è la variabile indipendente e l'equazione si considera risolta quando si trova il valore di Y che annulla W). Posto il valore logico del soggetto = 1 si ricava il valore del medio (procedendo per tentativi) e sostituendo il valore del medio secondo il procedimento già illustrato (ponendo cioè il valore del medio = 0, se 0 è il valore ricavato dalla prima equazione, = 1 in ogni altro caso) si ricava il valore del predicato. Ciò viene effettuato procedendo per tentativi (ponendo $Y=0$, poi $Y=0,5$ ed infine $Y=1$). A(3) ed A(4) contengono le eventuali soluzioni della prima premessa e della seconda. La variabile A contiene il numero di soluzioni di ciascuna equazione: se è $A > 1$ vuol dire che l'equazione è diventata un'identità

perdendo significato e sul display apparirà SCORRETTO. L'istruzione alla linea 78 chiama un sottoprogramma di correzione che, a seconda del tipo di proposizione della prima premessa, verifica se ci si trova in uno dei casi in cui l'algoritmo dà risultati errati. In tal caso sul display apparirà la dicitura SCORRETTO.

L'ultima parte del programma è dedicata alla traduzione dei risultati numerici sotto forma di proposizione conclusiva.

La conclusione sarà del tipo:

OGNI
QUALCHE S\$ (NON)
è P\$
NESSUN

Se il valore del termine medio (che è contenuto nella variabile N) è 0,5, la parola chiave sarà QUALCHE e NON comparirà a seconda che il valore del predicato (contenuto nella variabile Q) sia = 0 o no. Se il valore del predicato è 0 sul display apparirà NESSUNO, in ogni altro caso OGNI.

Conclusioni

L'aver insegnato al pocket a dedurre la corretta conclusione dalle premesse di un sillogismo può essere considerato, sotto un certo punto di vista, come una prima rudimentale forma di intelligenza artificiale.

Se consideriamo intelligenza la capacità di dedurre da certe premesse le conseguenze logiche (definizione che è certamente alquanto limitativa) potremmo dire che l'elaboratore è diventato "intelligente".

È difficile dire quali possano essere le applicazioni di un simile programma (a parte naturalmente l'uso che se ne può fare per sbalordire gli amici e per illuderli che il proprio computer sappia pensare): potrebbe forse essere di ausilio nello studio della logica oppure, qualora si operasse su elaboratori di dimensioni maggiori, potrebbe essere utile dedurre logicamente delle informazioni non contenute direttamente nell'archivio dati, ma contenute implicitamente come premesse da cui si possono dedurre le informazioni desiderate.

CONVERSIONI

SLOT-MACCHINE

Giochiamo d'azzardo sulle PC-1500

di E. Cima

Sul display vengono disegnati i frutti o anche gli altri oggetti di una normale slot machine. Quando sul lato sinistro del display viene raffigurato un punto, allora si potranno cambiare i semi (frutti o oggetti) per 5 volte, premendo i pulsanti sottostanti le figure. Quando invece, sempre sul lato sinistro del display, apparirà una riga trasversale, si potranno mantenere alcune figure simili in modo da vincere se escono in sequenza: le figure si mantengono premendo il pulsante sottostante

e a conferma apparirà un segno alla sinistra della figura tenuta. Per far procedere si preme il tasto "g". Se invece si è tenuta una figura sbagliata, premendo il tasto "s" si ripete questa sequenza.

Nel caso che si possano cambiare le figure, cioè solo quando compare il punto e si hanno già delle sequenze vincenti, si preme il tasto " ". Ogni volta che si preme il tasto "g" è come se si tirasse la leva della vera slot machine. Per giocare allo slot machine occorre avere l'espansione di memoria da 4K. ■

LISTATO "SLOT-MACHINE"

Il programma della slot-machine per la Sharp PC 1500.

Riproduce sul display a cristalli liquidi le figure e le combinazioni di una classica macchinetta da gioco d'azzardo.

```

100: WAIT 0: CLS :      BEEP 5: NEXT Y: 563: IF B=7 IF C=3 IF      GPRINT 24; 36
      CLEAR : DIM A(6   GOTO 559      D=5 IF E=8 IF F=      ; 90; 74; 90; 36
      )                530: IF D$=CHR$ &12 1LET KA=KA+800      ; 24
130: R$=INKEY$ :      LET C=RND 8:      0      1040: IF B=5
      WAIT 0           GOSUB 1090:      565: IF C=3 LET KA=K      GPCURSOR 10:
132: IF R$="G" GOTO    BEEP 5: NEXT Y:      A+100      GPRINT 8; 28;
      135             GOTO 559      567: IF F=1 LET KA=K      42; 119; 42; 28
133: GOTO 130          540: IF D$=CHR$ &13  A+100      ; 8
135: CLS : TA=TA+1:    LET D=RND 8:      571: GOSUB 4000      1050: IF B=6
      GPCURSOR 70:      GOSUB 1170:      572: IF KA=0 GOTO 58  GPCURSOR 10:
      PRINT TA:        BEEP 5: NEXT Y:      0      GPRINT 127; 6
      CURSOR 19:      GOTO 559      575: CLS : PRINT "HA  5; 73; 93; 73; 6
      PRINT K          541: IF D$=CHR$ &14  I VINTO"; KA; "  5; 127
140: B=INT (RND 8)    LET E=RND 8:      " : K=KA+K:      1060: IF B=7
150: C=INT (RND 8)    GOSUB 3000:      WAIT 0: FOR Y=0  GPCURSOR 10:
160: D=INT (RND 8):   BEEP 5: NEXT Y:      TO 50: BEEP 1, Y  GPRINT 8; 28;
      E=INT (RND 8):   GOTO 559      , 50: NEXT Y      41; 126; 41; 28
      F=INT (RND 8)    542: IF D$=CHR$ &15 580: GOTO 130      ; 8
200: GOSUB 1000:      LET F=RND 8:      1000: IF B=1      1070: IF B=8
      GOSUB 3000        GOSUB 3080:      GPCURSOR 10:      GPCURSOR 10:
430: P=RND 5          BEEP 5: NEXT Y:      GPRINT 32; 56      GPRINT 24; 36
440: IF P=1 THEN 200  GOTO 559      ; 62; 127; 62; 5  ; 66; 66; 66; 36
      0                545: IF D$="5" GOTO  6; 32      ; 24
450: IF P=3 THEN 470  559          1010: IF B=2      1090: IF C=1
460: GOTO 559          559: KA=0      GPCURSOR 10:      GPCURSOR 23:
470: WAIT 0           560: IF B=7 IF C=3  GPRINT 12; 28      GPRINT 24; 36
472: GPCURSOR 0:      LET KA=KA+1000  ; 61; 126; 61; 2  ; 90; 74; 90; 36
      GPRINT 48; 48    561: IF B=7 IF C=3 IF  8; 12      ; 24
490: FOR Y=1 TO 4     D=5 LET KA=KA+2  1020: IF B=3      1100: IF C=2
510: D$=INKEY$        000      GPCURSOR 10:      GPCURSOR 23:
520: IF D$=CHR$ &11   562: IF B=7 IF C=3 IF  GPRINT 8; 8; 2      GPRINT 24; 36
      LET B=RND 8:    D=5 IF E=8 LET K  8; 127; 28; 8; 8  ; 66; 66; 66; 36
      GOSUB 1000:      A=KA+3000      1030: IF B=4      ; 24
      GPCURSOR 10:

```


1110: IF C=3	GPRINT 8;28;	2100: GOTO 559	3120: IF F=5
GCURSOR 23:	42;119;42;28	2130: RETURN	GCURSOR 61:
GPRINT 8;28;	;8	3000: IF E=1	GPRINT 32;56
41;126;41;28	1245: IF D=8	GCURSOR 48:	;62;127;62;5
;8	GCURSOR 35:	GPRINT 12;28	6;32
1120: IF C=4	GPRINT 24;36	;61;126;61;2	3130: IF F=6
GCURSOR 23:	;90;74;90;36	8;12	GCURSOR 61:
GPRINT 127;6	;24	3010: IF E=2	GPRINT 127;6
5;73;93;73;6	1246: RETURN	GCURSOR 48:	5;73;93;73;6
5;127	2000: GCURSOR 0:	GPRINT 24;36	5;127
1130: IF C=5	GPRINT 127;0	;66;66;66;36	3140: IF F=7
GCURSOR 23:	;127:FOR I=1	;24	GCURSOR 61:
GPRINT 32;56	TO 5:A(I)=0:	3020: IF E=3	GPRINT 24;36
;62;127;62;5	NEXT I	GCURSOR 48:	;66;66;66;36
6;32	2001: AWS\$=INKEY\$	GPRINT 127;6	;24
1140: IF C=6	2010: IF AWS\$=CHR\$	5;73;93;73;6	3150: IF F=8
GCURSOR 23:	&11LET A(1)=	5;127	GCURSOR 61:
GPRINT 8;28;	1:GCURSOR 6:	3030: IF E=4	GPRINT 12;28
42;119;42;28	GPRINT 21	GCURSOR 48:	;61;126;61;2
;8	2020: IF AWS\$=CHR\$	GPRINT 32;56	8;12
1145: IF C=7	&12LET A(2)=	;62;127;62;5	3160: RETURN
GCURSOR 23:	1:GCURSOR 19	6;32	4000: IF B=6IF C=4
GPRINT 8;8;2	:GPRINT 21	3040: IF E=5	IF D=1IF E=3
8;127;28;8;8	2030: IF AWS\$=CHR\$	GCURSOR 48:	IF F=6LET KA
1150: IF C=8	&13LET A(3)=	GPRINT 24;36	=KA+60000
GCURSOR 23:	1:GCURSOR 32	;90;74;90;36	4010: IF B=6IF C=4
GPRINT 12;28	:GPRINT 21	;24	IF D=1IF E=3
;61;126;61;2	2040: IF AWS\$=CHR\$	3050: IF E=6	LET KA=KA+50
8;12	&14LET A(4)=	GCURSOR 48:	00
1170: IF D=1	1:GCURSOR 45	GPRINT 8;28;	4020: IF F=3LET KA
GCURSOR 35:	:GPRINT 21	42;119;42;28	=KA+100
GPRINT 127;6	2050: IF AWS\$=CHR\$	;8	4030: IF E=8IF F=1
5;73;93;73;6	&15LET A(5)=	3060: IF E=7	LET KA=KA+20
5;127	1:GCURSOR 58	GCURSOR 48:	0
1180: IF D=2	:GPRINT 21	GPRINT 8;8;2	4040: IF D=5IF E=8
GCURSOR 35:	2060: IF AWS\$="G"	8;127;28;8;8	IF F=1LET KA
GPRINT 32;56	GOTO 2080	3070: IF E=8	=KA+2000
;62;127;62;5	2065: IF AWS\$="S"	GCURSOR 48:	4050: IF C=3IF D=5
6;32	GOTO 2075	GPRINT 8;28;	IF E=8IF F=1
1190: IF D=3	2070: GOTO 2001	41;126;41;28	LET KA=KA+30
GCURSOR 35:	2075: FOR I=1TO 5:	;8	00
GPRINT 12;28	LET A(I)=0:	3080: IF F=1	4060: IF B=6IF C=4
;61;126;61;2	NEXT I:FOR X	GCURSOR 61:	IF D=1IF E=3
8;12	=6TO 58STEP	GPRINT 8;28;	LET KA=KA+10
1200: IF D=4	13:GCURSOR X	41;126;41;28	000
GCURSOR 35:	:GPRINT "00"	;8	4070: IF B=6IF C=4
GPRINT 8;8;2	:NEXT X	3090: IF F=2	IF D=1LET KA
8;127;28;8;8	2076: GOSUB 2001	GCURSOR 61:	=KA+5000
1210: IF D=5	2080: IF A(1)<>1	GPRINT 8;8;2	4080: IF C=4IF D=1
GCURSOR 35:	LET B=RND 8	8;127;28;8;8	IF E=3IF F=6
GPRINT 8;28;	2081: IF A(2)<>1	3100: IF F=3	LET KA=KA+10
41;126;41;28	LET C=RND 8	GCURSOR 61:	000
;8	2082: IF A(3)<>1	GPRINT 8;28;	4090: IF D=1IF E=3
1220: IF D=6	LET D=RND 8	42;119;42;28	IF F=6LET KA
GCURSOR 35:	2083: IF A(4)<>1	;8	=KA+5000
GPRINT 24;36	LET E=RND 8	3110: IF F=4	5000: IF B=1IF C=5
;66;66;66;36	2084: IF A(5)<>1	GCURSOR 61:	IF D=2LET KA
;24	LET F=RND 8	GPRINT 24;36	=KA+4500
1230: IF D=7	2085: GOSUB 1000:	;90;74;90;36	5010: IF B=1IF C=5
GCURSOR 35:	GOSUB 3000	;24	IF D=2IF E=4



UNA PUBBLICAZIONE
DEL GRUPPO EDITORIALE JACKSON

PERSONAL SOFTWARE

ANNO 2 N. 6 MAGGIO 1983

DIRETTORE RESPONSABILE: Giampietro Zanga

DIRETTORE: Mauro Boscarol

REDAZIONE: Paolo Capobussi e Completo Software - Padova

HANNO COLLABORATO A QUESTO NUMERO:

R. Amasino, E.M. Albani, S. Brown, W.L. Colsher,
F. Ferrari Bravo, E. Ferreguti, J. Grout, E. Mitchell,
M. Redolfi, F. Santini, M. Spero, S. Ventura.

Copertina e illustrazioni: MORGANA artefatti comunicativi - Padova
Fotocomposizione: Composizioni Grafiche - Padova

CONTABILITA': Franco Mancini, Roberto Ostelli,
Mariella Luciano, Franca Anelli, Sandra Cicuta,
Gabriella Napoli

DIFFUSIONE E ABBONAMENTI: Luigi De Cao,
Adela Bel Lozano, Ombretta Giannetto

AUTORIZZAZIONE ALLA PUBBLICAZIONE: Tribunale
di Milano n. 69 del 20/2/1982

PUBBLICITA': Concessionario per l'Italia e l'Estero
Reina s.r.l. Via Washington, 50 - 20146 Milano
Tel. (02) 4988066/7/8/9/060 (5 linee r.a.)
Telex 316213 REINA I

STAMPA: REWEBA - Brescia

Concessionario esclusivo per la DIFFUSIONE in Italia
e all'Estero:
SODIP - Via Zuretti, 25 - 20125 Milano

Spedizione in abbonamento Postale Gruppo III/70
Prezzo della rivista L. 3.500. Numero arretrato L. 6.000.
Abbonamento annuo (10 numeri) L. 30.000; per l'Estero
L. 48.000

I versamenti vanno indirizzati a: Gruppo Editoriale Jackson
Via Rosellini, 12 - 20124 Milano - mediante emissione di
assegno bancario, cartolina vaglia o utilizzando il c/c
Postale numero 11666203.

Per i cambi di indirizzo, indicare, oltre naturalmente al
nuovo, anche l'indirizzo precedente, ed allegare alla
comunicazione l'importo di L. 500, anche in francobolli.

© TUTTI I DIRITTI DI RIPRODUZIONE O TRADUZIONE
DEGLI ARTICOLI PUBBLICATI SONO RISERVATI



GRUPPO EDITORIALE JACKSON Srl

DIREZIONE, REDAZIONE, AMMINISTRAZIONE:
Via Rosellini, 12 - 20124 Milano - Telefoni: 68.03.68 - 68.00.54

SEDE LEGALE: Via Vincenzo Monti, 15 - 20123 Milano

DIREZIONE EDITORIALE: Giampietro Zanga e Paolo Reina
COORDINAMENTO EDITORIALE: Daniele Comboni

```

      LET KA=KA+80
      00
5020: IF B=1IF C=5
      IF D=2IF E=4
      IF F=5LET KA
      =KA+50000
5030: IF C=5IF D=2
      IF E=4IF F=5
      LET KA=KA+80
      00
5040: IF D=2IF E=4
      IF F=5LET KA
      =KA+4500
5050: IF B=5IF C=6
      IF D=7LET KA
      =KA+4000
5060: IF B=5IF C=6
      IF D=7IF E=6
      IF F=3LET KA
      =KA+50000
5070: IF C=6IF D=7
      IF E=6IF F=3
      LET KA=KA+75
      00
5080: IF D=7IF E=6
      IF F=3LET KA
      =KA+4500
5100: IF B=3IF C=7
      IF D=4LET KA
      =KA+4000
5110: IF B=3IF C=7
      IF D=4IF E=7
      LET KA=KA+75
      00
5120: IF B=3IF C=7
      IF D=4IF E=7
      IF F=2LET KA
      =KA+40000
5130: IF C=7IF D=4
      IF E=7IF F=2
      LET KA=KA+75
      00
5140: IF D=4IF E=7
      IF F=2LET KA
      =KA+4000
5150: IF B=2IF C=8
      LET KA=KA+50
      0
5160: IF B=2IF C=8
      IF D=3IF E=1
      LET KA=KA+10
      00
5170: IF B=2IF C=8
      IF D=3IF E=1
      IF F=8LET KA
      =KA+1500
5180: IF C=8IF D=3
      IF E=1IF F=8

      LET KA=KA+10
      00
5190: IF D=3IF E=1
      IF F=8LET KA
      =KA+800
5200: IF E=1IF F=8
      LET KA=KA+50
      0
5210: IF B=8IF C=2
      LET KA=KA+40
      0
5220: IF B=8IF C=2
      IF D=6LET KA
      =KA+700
5230: IF B=8IF C=2
      IF D=6IF E=2
      LET KA=KA+90
      0
5240: IF B=8IF C=2
      IF D=6IF E=2
      IF F=7LET KA
      =KA+1400
5250: IF C=2IF D=6
      IF E=2IF F=7
      LET KA=KA+90
      00
5260: IF D=6IF E=2
      IF F=7LET KA
      =KA+7000
5270: IF E=2IF F=7
      LET KA=KA+40
      0
5300: IF B=4IF C=1
      LET KA=KA+30
      0
5310: IF B=4IF C=1
      IF D=8LET KA
      =KA+600
5320: IF B=4IF C=1
      IF D=8IF E=5
      LET KA=KA+80
      0
5330: IF B=4IF C=1
      IF D=8IF E=5
      IF F=4LET KA
      =KA+1300
5340: IF C=1IF D=8
      IF E=5IF F=4
      LET KA=KA+80
      0
5350: IF D=8IF E=5
      IF F=4LET KA
      =KA+600
5360: IF E=5IF F=4
      LET KA=KA+30
      0
5370: RETURN
  
```


PICCOLI ANNUNCI

Commodore PET/CBM

Compro programma Sargon II per PET CBM 4032 su disco venduto da Hidden Book Company Inc. USA. Dott. Luciano Tursi, via Matteotti 10, 87023 Diamante (CS), tel. 0985-81113.

Cambio programmi per PET 4000: giochi (alcuni autoelaborati), grafica, bioritmi, vari. Carlo Pautaso, via Pastrengo 4, 10024 Moncalieri (TO), tel. 6060819.

Vendo per CBM 3032 telaio sismico su cassetta o disco completamente automatico. Involuppo delle azioni interne e calcolo armature minime in tutte le aste. Alvaro Albani, via Castelfidardo 7, 47037 Rimini (FO), tel. 0541-25765.

Cerco possessori CBM 64 per scambio di informazioni. Ringrazio chiunque possa segnalarmi riviste anche straniere, riguardanti il 64. Enrico Cortona, via Dante 6, 15062 Bosco Marengo (AL), tel. 0131-759119.

Vendo Acorn Aton 12/82 espanso a 16K ROM + 16K RAM + alimentatore + manuali istruzioni italiano e inglese cavetti + 2 Games originali, tutto a L. 700.000 valore reale - L. 880.650, ottime condizioni. Grazie. Agostino Bova, via F. Chelini 44, 55049 Stiava (Lucca), tel. 0584-92435.

Sinclair ZX80-ZX81 Spectrum

Scambio/vendo/compro programmi di qualsiasi genere per i computer Sinclair ZX81 e Spectrum per maggiori informazioni scrivere a Igor Bonat, via Rössmann 6, 34141 Trieste, tel. 768718.

Vendo ZX81+16K RAM + alimentatore + cavi + programmi (labirinto 3D, scacchi II° e tanti altri) + 4 libri sullo ZX81 tutto del dicembre '82 ed in ottime condizioni a 280.000. Bruno Lorenzi, via Amundsen 5, 20148 Milano, tel. 4032692.

Vendo per ZX81 programma Tartinville (per ricerca capisaldi in equazione parametrica di 2° grado) a L. 15.000 tutto compreso (cassetta + spese e spedizione) posseggo inoltre altri programmi interessanti. Bruno Cardella, via Calabria 4 Lotto 43, 90100 Palermo, tel. 512302.

Offro un fantastico poker per il tuo ZX81 o ZX80/8K! 4 cassette registrate zeppe di programmi 4K e 16K, una marea di giochi, progr. didattici e utilities. Chiedile, te le mando in visione gratis. Pagherai solo se sarai soddisfatto. Una L. 14.000, tutte e 4 L. 49.000 sconti e rateazioni ai giovani sinclairisti, decine di programmi 1K omaggio per tutti. Scrivi, sarai soddisfatto. Bruno Del Medico, via Torino 72, 04016 Sabaudia.

Club ZX81 forniamo 3 programmi provati su ZX81 in cambio di un programma inviatoci. Spediteci busta con vostro indirizzo affrancata + un bollo da L. 500 per indennizzo parziale fotocopiatura listati. Enio Solino, via Monza 42, 20047 Brugherio (MI), tel. 039-879145.

Vendo programmi per ZX81 cassetta comprendente defender 3D e simulazione volo L. 15.000 Invaders per ZX Spectrum L. 10.000 - L. 2.000 listino completo. Stefano Nocilli, via Giuseppe De Leva 23, 00179 Roma.

Vendo Sinclair ZX80 nuova ROM + vecchia ROM + alimentatore + cassetta programmi al migliore offerente. Prezzo base L. 100.000 + manuale vecchia ROM ideale per chi comincia! Emilio Triunfo, via Cumana 9, 80126 Napoli, 081-633247.

Korg sintetizzatore monofonico professionale MS-20 vendo a L. 600.000. ZX81 con libro in italiano, alimentatore 2, 5A, 16 Kbyte di RAM, cassette di scacchi e tirannosauro a L. 250.000. Luca Dal Passio, via D'Ossola 19, 20162 Milano, tel. 02-6435086.

Cambio/compro/vendo software di giochi per Sinclair ZX spectrum fondo a Milano Club di Spectri con biblioteca gratuita di programmi vari. Per informazioni telefonare ore serali (18.30-20). Claudio Zucca, via M. Pagano 35, 20145 Milano, tel. 4693299.

Vendo a prezzi modici software e hardware per ZX81 e Spectrum (anche programmi gestionali e scientifici). Scrivetemi e Vi invierò l'elenco con un programma regalo. Alberto Lanzoni, via F. Corridoni 20, 48022 Lugo (RA).

Cerco possessori del computer ZX Spectrum per scambio materiale software e informazioni - rispondi a tutti. "Spectrummisti" scrivete!! Riccardo Taccia, via Redi 53, 57100 Livorno, tel. 0586-853046.

Vendo programmi per ZX Spectrum in italiano (scacchi, ecc.) a prezzo molto basso; vendo inoltre libri inglesi sullo Spectrum (importante: sono in italiano) chiedere elenco. Emer Degani, via G. Luosi 204, 41100 Modena, tel. 350833.

Per ZX Spectrum 16 e 48K vendo o cambio oltre 40 cassette. Chiedere listino gratuito con modalità di scambio o vendita. Converto inoltre Spectrum 16K in 48K a L. 120.000, 1° e 2° edizione. Dante Vialotto, via Gorizia 5, 21053 Castellanza (VA), tel. 0331-500713.

Vendo ZX81 + espansione 16K + alimentatore + cavi collegamento + manuali istruzioni in italiano e inglese - in perfette condizioni negli imballi originali a L. 350.000 trattabili. Stefano Campostrini, via V. Veneto 3/A, 38068 Rovereto (Trento), tel. 0464-25711.

Vendo Sinclair ZX81 + alimentatore + espansione 16K + cavi collegamento TV e registratore + manuale originale ed in italiano + cassetta programmi e documentazioni varie. Franco Destefanis, via T. Calissano 6, 12051 Alba (CN), tel. 0173-2937.

Vendo Sinclair ZX81 + 16K RAM + alimentatore + manuali (italiano e inglese) il tutto a L. 300.000. Scrivere o telefonare ore cena a: Salvatore Grifeo, via M.B. Tosatti 26, 00137 Roma, tel. 06-8273671.

Compro nuova ROM (8K) per ZX80 o copia di detta, causa impossibilità reperimento per vie ufficiali per offerte scrivere a: Patrizio Temperanza, via Contigliano 20, 02100 Rieti, tel. 0746-40436.

Vendo ZX81 + 32K RAM + scheda musicale e porte I/O + Mother Board che consente di collegare fino a 5 schede allo ZX + alimentatore stabilizzato da 2,5 + software anche inedito tutto a L. 400.000. Sergio Roda, via Ciro Pollini 6, 37131 Verona, tel. 045-521333.

Vendo ZX81 16K + alimentatore + cavi, imballo originale, manuali in italiano e circa 200 listati di programmi (fra cui molti originali in inglese). Tutto a L. 400.000. Per accordi: Alessandro Jemma, via D. Sciallatti 3, 40141 Bologna, tel. 051-472134.

Vendo programmi per ZX81 fantastici tutti da 1K - giochi grafica - utility - contabilità, una cassetta con 12 programmi L. 12.000 - con 24 L. 20.000. Telefonare per informazioni: Paolo Narcisi, viale Riformazione 32, 10064 Pinerolo (TO), tel. 0121-71213.

Vendo ZX81 + alimentatore + manuali e vari programmi (Defender, scacchi, e più di 20 altri) a L. 200.000 per completare HI-FI. Francesco Gatti, via Lucio Elio Seiano 79, 00174 Roma, tel. 06-744257.

Vendo nuovi per ZX81 tastiera (tipo VIC 20) memotech interfaccia con uscita Post ZX L. 175.000 e HRG Memotech unità grafica alta Ris. L. 150.000 tutti e due nuovi, mobili d'alluminio errato acquisto. Jayne Donald, via Dei Bardi 34, 50100 Firenze, tel. 055-218095.

Apple II

Scambio programmi per Apple II, utilità, scientifici, gestionali, giochi. Mandami la tua lista che invierò la mia, per Toscana, Emilia R., Lombardia. Giorgio Becevel, via Di Ponzano 20, 50047 Prato (FI), tel. 0574-582564.

Cambio programmi per Apple II - cambio videogioco Philips G7000 con 9 cassette, con pari valore di programmi per Apple. Annuncio sempre valido per cambio-compera-vendita. Tristano Marchini, via Rosselli 6, 58033 Castel del Piano, tel. 0564-955549.

I lettori che voglio vendere, comperare o scambiare software, o desiderano dare informazioni possono compilare il tagliando pubblicato in fondo a questa rubrica. Il servizio è gratuito. La redazione si riserva il diritto insindacabile di rifiutare, sospendere o modificare qualsiasi inserzione.

Gli annunci sono riservati ai privati o ai club senza scopo di lucro.

Daremo la precedenza agli annunci che si riferiscono a software, programmi, libri e riviste, club per personal computer.

Un annuncio sarà più efficace se seguirete queste indicazioni:

- La *prima parola* deve essere esplicativa del vostro messaggio: scambio, vendo, compro, cerco... Per renderla più evidente la stamperemo in corsivo.
- Nel testo, riferitevi ad *un solo tipo* di computer (VIC 20, ZX80,...). L'annuncio apparirà sotto la testata relativa a quel computer. Se volete fare un annuncio per due o più computer, compilate due tagliandi.
- Date il vostro *recapito* con esattezza: nome e cognome, via e numero, *cap* e località, provincia, *prefisso* e numero telefonico.

Acquisto programmi di vario genere per VIC 20.
Inviare descrizioni dettagliate: Salvatore Piraneo,
via La Valle 16, 13062 Candelo (VC), tel. 015-
538254.

Vendo per Sharp MZ80A routine assembler per l'esecuzione di somme e sottrazioni algebriche con numeri interi di 15 cifre più segno. Consentono un notevole aumento di velocità nell'esecuzione dei programmi. Su richiesta anche con diverso numero di cifre. Piergiorgio Cresto, via Circonvallazione 27/3, 10018 Pavone Canavese, tel. 51480.

PADOVA Via Fistomba, 8
PALERMO Via Libertà, 191
PARMA Via Imbriani, 41
PAVIA Via C. Battisti, 4/A
PERUGIA Via R. D'Andreotto, 49/55
PESCARA Via Tiburtina, 264 bis
PESCARA Via Trieste, 73
PIACENZA Via IV Novembre, 60
PISA Via XXIV Maggio, 101
PISTOIA V.le Adua, 350
POTENZA Via G. Mazzini, 72
POZZUOLI Via G.B. Pergolesi, 13
PRATO Via E. Boni, 76/78
RIMINI Via Bertola, 75
ROMA L.go Belloni, 4 (Vigna Stelluti)
ROMA P.zza San Donà Di Piave, 14
ROMA V.le IV Vent, 152
ROMA Via Cerreto Da Spoleto, 23
ROMA Via Ponzo Cominio, 46
ROMA Via Del Traforo, 136
SAVONA Via G. Scarpa, 13/R
SONDRIO Via N. Sauro, 28
TERAMO Via Martiri Pennesi, 14
TERNI Via Beccaria, 20
TORINO C.so Grosseto, 209
TORINO Via Tripoli, 179
TORINO Via Nizza, 91
TRENTO Via Sighele, 7/1
TREVIGLIO V.le Buonarroti, 5/A
TRIESTE Via F. Saverio, 138
UDINE Via Tavagnacco, 89/91
VERESE Via Carrobbio, 13
VERCELLI Via Dionisotti, 18
VIAREGGIO Via A. Volta, 79
VOGHERA P.zza G. Carducci, 11

Città Tel.



ORA C'E'!

ZX Spectrum

- 16 o 48 kbytes RAM.
- grafica ad alta risoluzione (256x192 punti).
- 8 colori da utilizzare con la più assoluta libertà per testo, sfondo, bordo, in campo diretto o inverso, con due gradi di luminosità, a luce fissa o lampeggiante.
- Tastiera multifunzione con maiuscole, minuscole, simboli grafici, caratteri definibili dall'utente.
- BASIC Sinclair esteso con funzioni a un tasto per programmare in fretta e senza errori.
- Funzioni specifiche per la grafica e per la gestione di dati d'archivio.
- Ampia disponibilità di programmi preregistrati su compact-cassette: giochi, passatempi, educazionali, matematici, gestionali.
- Totale compatibilità con la stampante ZX.
- Disponibilità immediata del volume ALLA SCOPERTA DELLO ZX SPECTRUM in italiano.
- Prezzo eccezionale: 360.000 lire nella versione a 16 kbytes.



è distribuito da

REBIT
COMPUTER

A DIVISION OF G.B.C.

REBIT COMPUTER
Via Induno, 18
20092 CINISELLO BALSAMO
Casella Postale 10488 MI

Harden Italia. Il salto di qualità.

*Dal personal computer
al professional computer.*

Nel quadro di una filosofia aziendale in evoluzione, Harden Italia riconferma la validità della proposta del Sirius 1. Il Sirius 1, con tutta la potenza del suo microprocessore a 16 bit, con 5 MHz, e una memoria centrale che può arrivare a 896 KBytes, è uno dei più avanzati della nuova generazione dei Personal.

Oltre ad una enorme capacità di archiviazione dei dati (dai 1240 KBytes del Sirius 1 agli 11.840 KBytes del Sirius 1b) il Sirius può contare su alcune caratteristiche che un tecnico e un professionista non possono non apprezzare: dall'interfacciamento con due porte seriali e una parallela programmabile da software, ai sistemi operativi (MS-DOS della Microsoft e CP/M86 della Digital Research), fino ai linguaggi di alto livello come il BASIC-86 (interprete e compilatore), l'Assembler, il COBOL, il Fortran, il Pascal.

Oltre che sul software vero e proprio (programmi come il Dbase II, il SuperCalc, il Multiplan o l'Harden-text e l'Harden-data) il Sirius 1 si avvale dei così detti "Tool Kits", una serie cioè di utilities compatibili con qualsiasi linguaggio che permettono una stesura dei programmi più facile e più completa come ad esempio l'AutoSort, il FABS, una gestione sofisticata IS, ecc. In più, il Sirius 1 è distribuito e assistito dalla Harden Italia su tutto il territorio nazionale.

Per saperne di più sul Sirius 1, sui suoi programmi o su dove sono i punti di vendita Harden più vicini, chiamare (0372)-63136 oppure (02)-651645: risponde la Harden Italia.



**HI HARDEN
ITALIA**

Harden Italia S.p.A. Direzione generale e uffici commerciali
20121 Milano - via dei Giardini, 4 - tel. (02) 651645
Sede operativa e uffici commerciali
26048 Sospiro (CR) - tel. (0372) 63136 - telex: 3205881

SIRIUS 1 CONFIGURAZIONE BASE
(128 KBYTES RAM, 1240 KBYTES FLOPPY DISC)
DA OGGI L.6500.000